

**IMPLEMENTACIÓN DE UN ALGORITMO EVOLUTIVO PARA EL  
PROBLEMA DEL AGENTE VIAJERO (TSP)**

**FABIÁN ANDRÉS OCHOA GÁLVIS**



**UNIVERSIDAD PONTIFICIA BOLIVARIANA  
FACULTAD DE INGENIERÍA INDUSTRIAL  
SECCIONEAL BUCARAMANGA  
2014**

**IMPLEMENTACIÓN DE UN ALGORITMO EVOLUTIVO PARA EL  
PROBLEMA DEL AGENTE VIAJERO (TSP)**

**FABIÁN ANDRÉS OCHOA GÁLVIS**

**Investigación Aplicada de Pregrado para  
Optar al Título de Ingeniero Industrial**

**Docente Tutor**

**M.Sc. ORLANDO FEDERICO GONZÁLEZ CASALLAS**



**UNIVERSIDAD PONTIFICIA BOLIVARIANA  
FACULTAD DE INGENIERÍA INDUSTRIAL  
SECCIONEAL BUCARAMANGA  
2014**

**Nota de Aceptación:**

-----  
-----  
-----  
-----  
-----  
-----

-----  
**Firma del Presidente del Jurado**

-----  
**Firma del Jurado**

-----  
**Firma del Jurado**

**Bucaramanga, 2014.**

## **AGRADECIMIENTOS**

A los docentes universitarios GONZÁLEZ CASALLAS, Orlando Federico, MSc. y ACOSTA AMADO, Rolando, PhD. Quienes por vocación, mediante su orientación, enseñanza, paciencia, etc. Contribuyeron de manera significativa en la culminación exitosa de la presente investigación aplicada de pregrado.

A mis compañeros de estudio y pregrado por quienes de una u otra manera impulsaron la ejecución, éxito y culminación de este estudio de pregrado; a ellos:

AVENDAÑO LEAL, Diana Carolina

MEJÍA JÁLABE, Aizar

ANGARITA SEPULVEDA, Alvaro Javier

RICO SANCHEZ, Nicolás

CORREDOR C., Iván Mauricio

RINCÓN RODRÍGUEZ, Samara Andrea

ESCOBAR BUENO, Carlos Iván

RÍOS CRUZ, Kelly Johana

JAIMES SERRANO, Juan Mauricio

Les presento mi cordial respeto, un fuerte abrazo, mis mejores recuerdos y mis más buenos deseos.

Al semillero de investigación SIMOS de la Universidad Pontificia Bolivariana de Bucaramanga por liderar espacios que permiten la exploración, profundización y apropiación de conocimientos especializados en la solución de problemas de optimización.

A la Universidad Pontificia Bolivariana por ser la principal institución que guió y facilitó desde la semana de inducción hasta mi graduación en el pregrado de Ingeniería Industrial, aquellos recursos necesarios para la generación de valor en mi formación personal, social y profesional.

A los autores de la Bibliografía consultada.

## DEDICATORIA

A mi padre DIOS quien aún sin pedirle el maravilloso regalo de otro día más de vida, hasta este momento me ha permitido experimentar la gracia de su amor e inmensa misericordia en compañía de aquellos a los que amo.

A mi mamá JANETH MARGARETH GALVIS MACHUCA, el mejor apoyo incondicional que mi padre DIOS ha puesto en mi vida, fuente de mi inspiración para vivir agradecido por el resto de mis días.

A mi papá ADRIANO OCHOA GÓMEZ, por su amor, apoyo moral y económico, preocupación y exhortación en la consecución de mis metas personales, y por la presencia de su vida en mi vida.

A mis hermanos de sangre: NATALIA PAOLA FORERO GALVIS, JEYSON ADRIÁN OCHOA GALVIS y OSCAR DAVID OCHOA CUJAR, mis mejores amigos a quienes amo sinceramente para siempre.

A mi abuela materna TELVANIA MARÍA MACHUCA URIETA, en quien veo a mi otra mamá.

A mi familia putativa: Sr. JOSE MANUEL GARCÍA MARTINEZ, Sra. GLORIA ELVIRA SANTISTEBAN ALVARADO, e hijos ANDRÉS FABIÁN GARCÍA SANTISTEBAN, JOSE MIGUEL GARCÍA SANTISTEBAN y KAREN GARCÍA SANTISTEBAN, por quienes he descubierto el valor real de la amistad.

A ADELAIDE HERCILIA CUJAR TORRES, esposa de mi papá y madre de mi hermano OSCAR; una gran amiga y consejera.

## ÍNDICE GENERAL

	Pág.
<b>ÍNDICE GENERAL</b> .....	<b>6</b>
<b>LISTA DE ILUSTRACIONES</b> .....	<b>10</b>
<b>LISTA DE GRÁFICAS</b> .....	<b>11</b>
<b>LISTA DE TABLAS</b> .....	<b>12</b>
<b>INTRODUCCIÓN</b> .....	<b>16</b>
<b>1. DEFINICIÓN DEL PROBLEMA</b> .....	<b>17</b>
<b>2. ANTECEDENTES</b> .....	<b>19</b>
2.1 HISTORIA DEL TSP .....	19
2.2 ANTECEDENTES DEL AG Y LA SOLUCIÓN APROXIMADA DEL TSP .....	20
2.2.1 <i>Clasificación de la Literatura Consultada</i> .....	20
2.2.1.1 <i>Estrategias para Evitar la Prematura Convergencia de la Población de Tratamiento</i> .....	20
2.2.1.2 <i>Modificación de los Operadores de Cruce como Estrategia de Diversificación de la Población de Tratamiento</i> .....	20
2.2.1.3 <i>Análisis de los Valores de los Parámetros de Entrada para un Buen Funcionamiento del Metaheurístico A.G.</i> .....	20
2.2.1.4 <i>Implementación de Variantes del AG para la Solución del TSP</i> .....	21
2.2.1.5 <i>Implementación de Metaheurísticos de Solución para la Resolución TSP</i> .....	21
2.2.2 <i>Estrategias para Evitar la Prematura Convergencia de la Población de Tratamiento</i> . 21	
2.2.2.1 <i>Mechanisms to Avoid the Premature Convergence of Genetic Algorithms; por NICOARĂ, Elena Simona. [38]</i> .....	21
2.2.2.2 <i>Reducing Premature Convergence Problem in Genetic Algorithm_ Application on Travel Salesman Problem; por ZEYAD RAMADAN, Saleem. [44]</i> .....	22
2.2.3 <i>Modificación de los Operadores de Cruce como Estrategia de Diversificación de la Población de Tratamiento</i> .....	22
2.2.3.1 <i>Developing Improved Greedy Crossover to Solve Symmetric Traveling Salesman Problem; por ISMKHAN, Hassan; ZAMANIFAR, Kamran. [27]</i> .....	22
2.2.3.2 <i>Genetic Algorithm for the Traveling Salesman Problem Using Sequential Constructive Crossover Operator; por AHMED H. Zakir. [1]</i> .....	22
2.2.4 <i>Análisis de los Valores de los Parámetros de Entrada para un Buen Funcionamiento del Metaheurístico A.G.</i> .....	23
2.2.4.1 <i>Analysis of the Impact of Parameters Values on the Genetic Algorithm for TSP; por DIKA, Agni; MAXHUNI, Adnan; REXHEPI, Avni. [22]</i> .....	23
2.2.4.2 <i>Optimal Mutation and Crossover Rates for a Genetic Algorithm Operating in a Dynamic Environment; por DAIDA M., Jason. STANHOPE A., Stephen. [15]</i> .....	23
<b>3. JUSTIFICACIÓN</b> .....	<b>24</b>
<b>4. OBJETIVOS</b> .....	<b>25</b>
4.1 OBJETIVO GENERAL.....	25

4.2	OBJETIVOS ESPECÍFICOS.....	25
<b>5.</b>	<b>MARCO TEÓRICO.....</b>	<b>26</b>
5.1	INVESTIGACIÓN DE OPERACIONES .....	26
5.1.1	<i>Origen de la Investigación de Operaciones .....</i>	26
5.1.2	<i>Definición de Investigación de Operaciones.....</i>	26
5.2	TEORÍA DE LA COMPLEJIDAD COMPUTACIONAL.....	27
5.2.1	<i>Definición de la Teoría de la Complejidad Computacional .....</i>	27
5.2.2	<i>Clasificación NP-Hard del TSP. ....</i>	28
5.3	PROBLEMA DEL AGENTE VIAJERO.....	28
5.3.1	<i>Definición del Problema del Agente Viajero Simétrico (sTSP) .....</i>	28
5.3.2	<i>Complejidad .....</i>	28
5.3.3	<i>Formulación Matemática del TSP.....</i>	29
5.3.4	<i>Variantes del TSP .....</i>	30
5.4	METAHEURÍSTICOS .....	30
5.4.1	<i>Definición de Metaheurísticos .....</i>	30
5.4.2	<i>Clasificación de Metaheurísticos de Búsqueda .....</i>	31
5.4.3	<i>Métodos Heurísticos .....</i>	32
5.5	ALGORITMOS GENÉTICOS.....	32
5.5.1	<i>Definición de los Algoritmos Genéticos .....</i>	32
5.5.2	<i>Operadores Genéticos de los AGs .....</i>	33
5.5.2.1	<i>Selección .....</i>	33
5.5.2.2	<i>Cruzamiento.....</i>	33
5.5.2.3	<i>Mutación .....</i>	33
5.5.3	<i>Pseudocódigo General del Algoritmo Genético .....</i>	33
5.5.4	<i>Aplicaciones Recientes de los Algoritmos Genéticos.....</i>	34
5.5.5	<i>Estructura Equivalente entre la Genética y el AG.....</i>	34
5.5.5.1	<i>Genotipo .....</i>	34
5.5.5.2	<i>Fenotipo.....</i>	34
5.5.5.3	<i>Cromosoma.....</i>	34
5.5.5.4	<i>Gen .....</i>	35
5.5.5.5	<i>Alelo.....</i>	35
5.5.5.6	<i>Función de aptitud.....</i>	35
5.6	HEURÍSTICO DE AHORROS DE CLARKE AND WRIGHT .....	35
5.6.2	<i>Algoritmo de Ahorros Clarke &amp; Wright. ....</i>	36
<b>6.</b>	<b>DISEÑO METODOLÓGICO .....</b>	<b>37</b>
<b>7.</b>	<b>UN ALGORITMO EVOLUTIVO PARA EL PROBLEMA DEL AGENTE VIAJERO ..</b>	<b>38</b>
7.1	ASPECTOS RELEVANTES Y ANÁLISIS DEL PSEUDOCÓDIGO DEL AG PARA SU IMPLEMENTACIÓN EN LA SOLUCIÓN APRÓX. DEL sTSP .....	38
7.1.1	<i>Pseudocódigo General del AG Propuesto (AG-sTSP). ....</i>	38
7.1.2	<i>Etapas 1: Generación de la Población de Tratamiento. ....</i>	39
7.1.2.1	<i>Representación, Codificación de Individuos .....</i>	39
7.1.2.2	<i>Tamaño de la Población de Tratamiento .....</i>	39
7.1.2.3	<i>Generación de la Población Inicial de Tratamiento.....</i>	39
7.1.3	<i>Etapas 2: Evaluación de la Población de Tratamiento. ....</i>	40

7.1.3.1	<i>Convergencia Prematura del AG</i> .....	40
7.1.3.2	<i>Definición de la Función Aptitud</i> .....	40
7.1.3.3	<i>Proceso de Selección Natural</i> .....	41
7.1.4	<i>Etapa 3: Evolución de la Población de Tratamiento</i> .....	41
7.1.4.1	<i>Criterio de Parada</i> .....	41
7.1.4.2	<i>Proceso de Cruce</i> .....	41
7.1.4.3	<i>Proceso de Mutación</i> .....	42
7.1.4.4	<i>Proceso de Inserción Élite</i> .....	42
7.2	<b>DISEÑO DE UN ALGORITMO GENÉTICO PARA LA RESOLUCIÓN DE sTSP</b> .	43
7.2.1	<i>Codificación de los Parámetros de Entrada del AG-sTSP</i> .....	43
7.2.2	<i>Etapa 1: Generación de la Población de Tratamiento</i> .....	43
7.2.2.1	<i>Individuo Semilla “Clarke &amp; Wright”</i> .....	43
7.2.2.2	<i>Alteración Probabilística del Individuo Semilla “BIT-SPLIT”</i> .....	44
7.2.3	<i>Etapa 2: Evaluación de la Población de Tratamiento</i> .....	45
7.2.3.1	<i>Evaluación de la Población de Tratamiento</i> .....	46
7.2.3.2	<i>Identificación de la Población Élite Inicial</i> .....	46
7.2.3.3	<i>Selección Estocástica de Supervivientes de la Población de Tratamiento</i> .....	47
7.2.4	<i>Etapa 3: Evolución de la Población de Tratamiento</i> .....	47
7.2.4.1	<i>Cruce Probabilístico de Individuos de la Población de Tratamiento</i> .....	47
7.2.4.1.1	<i>Cruce Probabilístico por Cruce Cíclico Ordenado</i> .....	49
7.2.4.1.2	<i>Cruce Probabilístico por Cruce Ordenado Multipunto</i> .....	50
7.2.4.2	<i>Mutación SIM de la Nueva Población de Tratamiento Restante</i> .....	52
7.2.4.3	<i>Inserción de Individuos Élite sobre la Nueva Población de Tratamiento</i> .....	53
7.2.4.3.1	<i>Identificación de la Nueva Población de Élite e Inadaptados</i> .....	54
7.2.4.3.2	<i>Comparación de las Poblaciones Élite</i> .....	55
7.3	<b>IMPLEMENTACIÓN DEL AG DISEÑADO EN LA RESOLUCIÓN DEL sTSP</b> .....	56
7.3.1	<i>Experimentación</i> .....	56
7.3.1.1	<i>Características del Ordenador</i> .....	57
7.3.1.2	<i>Instancias sTSP-TSPLIB</i> .....	57
7.3.1.3	<i>Identificación de Factores</i> .....	57
7.3.1.4	<i>Número de Experimentos</i> .....	58
7.3.2	<i>Resultados Experimentales</i> .....	58
7.3.2.1	<i>Instancia bays29</i> .....	60
7.3.2.2	<i>Instancia eil51</i> .....	61
7.3.2.3	<i>Instancia st70</i> .....	62
7.3.2.4	<i>Instancia eil76</i> .....	64
7.3.2.5	<i>Instancia pr76</i> .....	65
7.3.3	<i>Análisis General de Resultados del Procesamiento AG-sTSP</i> .....	66
7.3.4	<i>Diseño de Experimentos (DOE) de Efectos Fijos para la Instancia bays29</i> .....	68
7.3.5	<i>Análisis de Resultados DOE Efectos Fijos Según B&amp;B para la Instancia bays29</i> .	70
7.3.6	<i>Recomendación DOE Efectos Fijos Según B&amp;B para la Instancia bays29</i> .....	71
7.3.7	<i>Tiempos de Procesamiento Promedio de Instancias sTSP-TSPLIB</i> .....	71
7.3.8	<i>Tiempos de Procesamiento Promedio Ts-TSP UPB Vs. AG-sTSP</i> .....	73
7.4	<b>IMPLEMENTACIÓN DEL MÉTODO DE SOLUCIÓN EXACTA (MSE) EN LA RESOLUCIÓN DEL sTSP</b> .....	74
7.4.1	<i>Tiempo Promedio de Procesamiento de Instancias sTSP-TSPLIB con MSE</i> .....	75
7.4.2	<i>Análisis General de Resultados del Procesamiento MSE</i> .....	76
	<b>LIMITACIONES</b> .....	<b>77</b>

<b>CONCLUSIONES</b> .....	<b>78</b>
<b>RECOMENDACIONES</b> .....	<b>80</b>
<b>BIBLIOGRAFÍA</b> .....	<b>82</b>
<b>WEBGRAFÍA</b> .....	<b>87</b>
<b>ANEXOS</b> .....	<b>88</b>
ANEXO A.....	89
ANEXO B.....	91
ANEXO C.....	92
ANEXO D.....	96
ANEXO E.....	97
ANEXO F.....	98
ANEXO G.....	112
ANEXO H.....	113
ANEXO I.....	114

## LISTA DE ILUSTRACIONES

	<b>Pág.</b>
Ilustración 1. Juego Icosiano de Hamilton.....	19
Ilustración 2. Pseudocódigo del AG.....	33
Ilustración 3. Algoritmo de Ahorros C & W.....	36
Ilustración 4. Método Científico.....	37
Ilustración 5. Pseudocódigo General del AG Propuesto (AG-sTSP).....	38
Ilustración 6. Representación, Codificación del individuo para TSP.....	39
Ilustración 7. Árbol B&B para la Configuración de Parámetros de Entrada en bays29.....	70
Ilustración 8. George Dantzig, Ray Fulkerson, and Selmer Johnson (1954); n=49.....	89
Ilustración 9. Procter and Gamble Ran a Contest in 1962; n=33.....	89
Ilustración 10. Groetschel (1977); n=120.....	89
Ilustración 11. Padberg and Rinaldi (1987); n=538.....	89
Ilustración 12. Groetschel and Holland (1987); n=666.....	89
Ilustración 13. Padberg and Rinaldi (1987); n=2392.....	89
Ilustración 14. Applegate, Bixby, Chvátal, and Cook (1994); n=7397.....	90
Ilustración 15. Applegate, Bixby, Chvátal, and Cook (1998); n=13509.....	90
Ilustración 16. Applegate, Bixby, Chvátal, and Cook (2001); n=15112.....	90
Ilustración 17. Applegate, Bixby, Chvátal, Cook, and Helsgaun (2004); n=24978.....	90
Ilustración 18. Pseudocódigo Gral. del operador de cruce propuesto "OCX".....	113
Ilustración 19. Pseudocódigo Gral. del operador de cruce propuesto "MOX".....	114

## LISTA DE GRÁFICAS

	<b>Pág.</b>
Gráfica 1. Comportamiento de los Rangos de Resultados Factibles en bays29. ....	60
Gráfica 2. Comportamiento de los Rangos de Resultados Factibles en eil51. ....	62
Gráfica 3. Comportamiento de los Rangos de Resultados Factibles en st70. ....	63
Gráfica 4. Comportamiento de los Rangos de Resultados Factibles en eil76. ....	65
Gráfica 5. Comportamiento de los Rangos de Resultados Factibles en pr76. ....	66
Gráfica 6. Tiempos Promedios de Procesamiento AG-sTSP Vs. Ts-TSP UPB. ....	74
Gráfica 7. Resultados de la Corrida 1, Réplica 1 de la Sesión bays29 (09).....	112

## LISTA DE TABLAS

	Pág.
Tabla 1. Variantes del Problema del Agente Viajero (TSP).....	30
Tabla 2. Clasificación de Metaheurísticos de Búsqueda. ....	31
Tabla 3. Similitud Conceptual entre la Genética y AGs. ....	34
Tabla 4. Instancias sTSP de TSPLIB Procesadas en el AG-sTSP.....	57
Tabla 5. Variables para el Arreglo Experimental Factorial $2^k$ Completo. ....	58
Tabla 6. Referencias de Costos de Rutas Factibles de bays29. ....	60
Tabla 7. Rangos de Resultados Factibles de cada Sesión en la Instancia bays29. ....	60
Tabla 8. Referencias de Costos de Rutas Factibles de eil51. ....	61
Tabla 9. Rangos de Resultados Factibles de cada Sesión en la Instancia eil51. ....	61
Tabla 10. Referencias de Costos de Rutas Factibles de st70. ....	62
Tabla 11. Rangos de Resultados Factibles de cada Sesión en la Instancia st70. ....	62
Tabla 12. Referencias de Costos de Rutas Factibles de eil76. ....	64
Tabla 13. Rangos de Resultados Factibles de cada Sesión en la Instancia eil76. ....	64
Tabla 14. Referencias de Costos de Rutas Factibles de pr76.....	65
Tabla 15. Rangos de Resultados Factibles de cada Sesión en la Instancia pr76.....	65
Tabla 16. Configuración Recomendada de los Parámetros de Entrada AG-sTSP para bays29.....	71
Tabla 17. Tiempo Promedio de Procesamiento de Instancias TSPLIB con Ts-TSP UPB.....	72
Tabla 18. Tiempo Promedio de Procesamiento AG-sTSP de bays29. ....	72
Tabla 19. Tiempo Promedio de Procesamiento AG-sTSP de eil51 Parte 1.....	72
Tabla 20. Tiempo Promedio de Procesamiento AG-sTSP de eil51 Parte 2.....	72
Tabla 21. Tiempo Promedio de Procesamiento AG-sTSP de st70.....	73
Tabla 22. Tiempo Promedio de Procesamiento AG-sTSP de eil76.....	73
Tabla 23. Tiempo Promedio de Procesamiento AG-sTSP de pr76. ....	73
Tabla 24. Tiempos Promedios de Procesamiento de instancias TSPLIB con MSE y MOM. ....	75
Tabla 25. Algunas Aplicaciones de la Investigación de Operaciones.....	91
Tabla 26. Aplicaciones de los Algoritmos Genéticos en la Optimización y Búsqueda. ....	92
Tabla 27. Probabilidades de Cruce y Mutación en la Literatura. ....	96
Tabla 28. Parámetros de Entrada del Algoritmo Genético Propuesto (AG-sTSP). ....	97
Tabla 29. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia bays29.....	98
Tabla 30. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia bays29.....	98
Tabla 31. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia eil51.....	99
Tabla 32. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia eil51.....	99

Tabla 33. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia st70.....	100
Tabla 34. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia st70. ....	101
Tabla 35. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia eil76.....	103
Tabla 36. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia eil76.....	103
Tabla 37. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia pr76. ....	105
Tabla 38. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia pr76. ....	106

## RESUMEN GENERAL DEL TRABAJO DE GRADO

**TÍTULO:** Implementación de un algoritmo evolutivo para el problema del agente viajero (tsp)

**AUTOR:** Fabián Andrés Ochoa Gálvis

**FACULTAD:** Ingeniería Industrial

**DIRECTOR:** Orlando Federico González Casallas

De acuerdo a su gran aplicabilidad en el ámbito comercial, industrial y académico, el problema del agente viajero demuestra ser uno de los problemas de optimización combinatoria mayormente estudiados por la comunidad científica en el campo de la investigación de operaciones, a tal fin, que últimamente ha venido siendo considerado como una prueba casi obligatoria para la validación de cualquier técnica de resolución de problemas enteros o combinatorios. En la presente investigación aplicada de pregrado, para hallar una solución aproximada al problema del agente viajero en su variante simétrica, se implementó el metaheurístico de búsqueda dirigida, el "Algoritmo Genético" para el procesamiento de un conjunto de 5 instancias TSPLIB utilizadas también en el desarrollo de la investigación "Un enfoque de búsqueda tabú por Jhon Gabriel" para efecto de comparación de resultados. Lo anterior, se logró mediante la inclusión de conceptos ajenos a su pseudocódigo tradicional, permitiendo así, la adopción de un modelo de búsqueda intensivo y exploratorio, donde entre estos, dichos conceptos fueron: 1. Individuo semilla o ancestro común para la generación de una población inicial de soluciones, 2. Población élite para el almacenamiento temporal de la información genética de las mejores respuestas, 3. Población inadaptada para la identificación y eliminación de las soluciones que degradan la calidad de las soluciones de la población en el proceso de evolución y 4. Utilización probabilística de dos operadores de cruce modificados. Además de que se evaluó cualitativamente el desempeño de dicho algoritmo evolutivo de acuerdo a la programación lineal del problema codificada GAMS, también se emitió para la instancia bays29 una configuración recomendada para los parámetros de entrada del Algoritmo Genético propuesto a partir del desarrollo de un diseño de experimentos de efectos fijos 2 a la k replicado bajo el enfoque de la metodología Branch & Bound.

**PALABRAS CLAVE:** Metaheurístico, Algoritmo Genético, Problema del agente viajero, Investigación de operaciones.

## GENERAL SUMMARY OF WORK OF GRADE

**TÍTULO:** Implementation of an Evolutive Algorithm for the traveling salesman problem (tsp)

**AUTHOR:** Fabián Andrés Ochoa Gálvis

**FACULTY:** Ingeniería Industrial

**DIRECTOR:** Orlando Federico González Casallas

According to its wide applicability in the trade, industry and academia, the traveling salesman problem proves to be one of the combinatorial optimization problems mostly studied by the scientific community in the operations research field, to that point, which ultimately it has been being considered as an almost mandatory testing for validation of any combinatorial or integer problems resolution technique. In this applied research, to find an approximate solution to the traveling salesman problem in its symmetric variant, the directed search metaheuristic implemented, was the "Genetic Algorithm" for processing a set of 5 instancias TSPLIB also used in the research development "a tabu search approach by John Gabriel" for purposes of benchmarking. This was achieved by including concepts beyond their traditional pseudocode, allowing the adoption of a model of an exploratory and intensive search, where among these, those concepts were: 1. Seed individual or common ancestor for the generation of an initial population of solutions, 2. Elite Population for temporary storage of genetic information from the best answers, 3. Unsuitable population for identifying and removing solutions that degrade the quality population of the solutions of the population in the process of evolution and 4. Probabilistic using two modified crossover operators. Also the performance of this evolutionary algorithm was qualitatively evaluated of the based on linear programming encoded problem GAMS, also was issued for the instance bays29 a recommended input parameters of the Genetic Algorithm proposed through the development of a replicated design of experiments 2 k under the focus of the Branch & Bound methodology.

**PALABRAS CLAVE:** Metaheuristic, Genetic Algorithm, the traveling salesman problem, Operations Research.

## INTRODUCCIÓN

De origen militar en la II guerra mundial, la investigación de operaciones es una disciplina que se ocupa del diseño y aplicación de métodos analíticos avanzados para la toma de “mejores” decisiones, empleando técnicas de otras ciencias matemáticas como el modelamiento matemático, análisis estadístico, optimización matemática, etc.; de aquí los conceptos de minimización y maximización, los cuales se han convertido en los pilares que fundamentan su funcionalidad, vigencia y desarrollo tecnológico hasta los presentes días, dando respuesta a problemas complejos que se presentan en la realidad cotidiana de las sociedades entre los cuales se destaca el problema del agente viajero (TSP) que tendrá especial tratamiento en la presente investigación aplicada de pregrado.

Dada la naturaleza compleja (NP-Hard) de algunos problemas de optimización como la del TSP [inciso 5.2.2], existen técnicas computacionales especializadas -Metaheurísticos- para la resolución de problemas que explotan estrategias de búsqueda de mejores soluciones mediante procedimientos genéricos y abstractos. Este es el caso de una solución propuesta para el Problema de Rutas de Vehículos con Capacidad (CVRP) en Medellín-Colombia<sup>1</sup> en el que sus autores logran mejorar la planeación y asignación de las rutas de transporte disminuyendo el costo fijo de la capacidad instalada en un 50%; otro, es el caso de la implementación de un algoritmo genético-inmune para el TSP<sup>2</sup>, en la que se empleó el concepto de inoculación de genes para mejorar la adaptación o aptitud de las soluciones candidatas al problema, reduciendo el tiempo computacional hacia la obtención de la mejor solución aproximada; y por mencionar uno último, es el caso de una investigación desarrollada en Chile que tuvo como fin, el perfeccionamiento de heurísticos para el TSP<sup>3</sup> mediante la implementación de indicadores de trazabilidad en el algoritmo-base utilizado, incluyendo así, etapas de detección e intervención del mismo para su posterior mejoramiento.

Al darse una muestra de la aplicabilidad y ventaja de la utilización de Metaheurísticos sofisticados para la resolución de problemas NP-Hard, en el presente se implementará un Algoritmo Genético (AG) para hallar una solución aproximada al problema del agente viajero en su variante simétrica, aportando en el área de la investigación de operaciones la estructuración del conocimiento adquirido como cimientos que guíen e incentiven a una nueva generación de estudiantes universitarios en la decisión de incursionar en la profundización y utilización de estas técnicas para el desarrollo de investigaciones futuras planteando soluciones rentables para la sociedad.

---

<sup>1</sup> DAZA, Julio Mario; MONTOYA, Jairo R.; NARDUCCI, Francesco. Resolución del Problema de Enrutamiento de Vehículos con Limitaciones de Capacidad (CVRP) Utilizando un Procedimiento Metaheurístico de dos fases. Medellín. Diciembre, 2009. ISSN 1794-1237. [Citados el 14 de Noviembre de 2013]

<sup>2</sup> JINGUI Lu; MIN, Xie. Immune-Genetic Algorithm for Traveling Salesman Problem, Traveling Salesman Problem, Theory and Applications. December, 2010. p. 81-96. ISBN 978-953-307-426-9.

<sup>3</sup> COCKBAINE OJEDA, Juan; SILVA URREA, Rubén. Perfeccionando algoritmos heurísticos para el problema NP-C E-TSP. Revista chilena de ingeniería, vol. 21 N° 2, 2013, p. 196-204

## 1. DEFINICIÓN DEL PROBLEMA

El problema objeto de estudio de la presente investigación, como desarrollo tecnológico y experimental, se encuentra enmarcado en el terreno de la optimización combinatoria dentro del contexto de la investigación de operaciones, y ha sido clasificado por Karp en 1972 como un problema de naturaleza Np-Hard (no polinomial difícil, por sus siglas en inglés) según la teoría de complejidad computacional. Esta clasificación ha sido atribuida a los problemas de optimización que, debido a la dificultad exponencial que presentan, no existe un *algoritmo determinista* que pueda resolverlos en un tiempo polinomial razonable.

El problema se aborda como el del agente viajante en su variante simétrica, tan conocido en el inglés como *The Symmetric Traveling Salesman Problem (sTSP)*, el cual, de acuerdo a la simplicidad que presenta su formulación pero complejidad NP-Hard en su solución, ha tomado la forma de un espécimen de laboratorio, “considerándose como una prueba obligatoria para la validación de cualquier técnica de resolución de problemas enteros o combinatorios”<sup>4</sup>. Para su contextualización, el caso del TSP puede definirse de un modo sencillo en la siguiente manera:

Asumiendo que un vendedor viajero (Traveling Salesman) tiene que visitar un número determinado de ciudades una sola vez cada ciudad, antes de emprender su viaje, éste debe establecer un secuencia de visitas para las ciudades en cuestión teniendo en cuenta que el lugar de su partida inicial a su vez será el lugar de destino final; entonces, dicha cadena o arreglo de visitas describirá un tour que representa una longitud de trayecto a recorrer, para el cual, lo que se busca en el campo de la optimización, es disminuir la razón de costo atribuida a dicho recorrido (entendiéndose costo como distancia, valor monetario, tiempo o cualquier otra variable que guarde proporcionalidad con el recorrido).

Como se menciona anteriormente, en la presente investigación será tratado para su solución el TSP en su variante simétrica (sTSP), el cual posee una característica fundamental descrita entre las distancias de visitas del tour, donde: “*La distancia que se recorra desde la ciudad  $i$  a la ciudad  $j$  es equivalente a la distancia que se recorra desde la ciudad  $j$  a la ciudad  $i$* ”.

$$d_{ij} = d_{ji}$$

Los recursos con los que se dispone inicialmente para el desarrollo de la presente investigación aplicada de pregrado son:

---

<sup>4</sup> MARTÍ, Rafa. Procedimientos Metaheurísticos en Optimización Combinatoria. [En Línea] Disponible en Pdf. URL: <<http://www.uv.es/rmart/paper/docs/heur1.pdf>> Universidad de Valencia, Facultad de matemáticas, departamento de estadística e investigación operativa p. 8 [Citado el 14 de Noviembre de 2013]

- ❖ Información de la ubicación espacial en 2D de las ciudades (coordinadas euclidianas y/o matriz simétrica de distancias) que agrupadas, conforman las instancias sTSP disponibles en la librería de domino publico TSPLIB<sup>5</sup>.
- ❖ Se cuenta con el arreglo de visitas o tour óptimo para la instancia simétrica del sTSP que se trate, para su posterior análisis y comparación con los resultados experimentales obtenidos de las investigaciones de pregrado “Un enfoque de búsqueda tabú para el TSP (UPB – Bucaramanga - Colombia)” y “la presente”.
- ❖ Disposición del software de “Sistema de Modelado Algebraico General” (GAMS)<sup>6</sup> para la programación y modelado de problemas de optimización y matemáticos de gran magnitud y complejidad tanto lineales, no lineales o de enteros mezclados.
- ❖ Disposición de Material bibliográfico de carácter científico en las base de datos EBSCO, SCIELO<sup>7</sup>, revistas electrónicas internacionales (Journals) SCIENTIFIC RESEARCH *open Access*<sup>8</sup>, GLOBETHICS<sup>9</sup>, INFORMS.PUBSONLINE<sup>10</sup>, CAMBRIDGE UNIVERSITY PRESS<sup>11</sup>, por mencionar algunos.
- ❖ Tutoría y dirección profesional de docentes UPB Bucaramanga altamente calificados, a través del ejercicio del semillero de investigación SIMOS de la universidad Pontificia Bolivariana de Bucaramanga - Colombia.

---

<sup>5</sup> TSPLIB. Symmetric Traveling Salesman Problem (sTSP). Public Domain Library. [On Line] URL: <<http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/tsp/>>

<sup>6</sup> GAMS. Home Page. [On Line] URL: <<http://www.gams.com/>>

<sup>7</sup> Scielo, Scientific Electronic Library Online. [On Line] URL: <[www.scielo.org](http://www.scielo.org)>

<sup>8</sup> SCIENTIFIC RESEARCH open Access. [On Line] URL: <<http://www.scirp.org/journal/OpenAccess.aspx>>.

<sup>9</sup> GLOBETHICS. [On Line] URL: <<http://www.globethics.net/>>

<sup>10</sup> Informa.Pubsonline. [On Line] URL: <<http://pubsonline.informs.org/journal/opre>>

<sup>11</sup> Cambridge Journals. [On Line] URL: <<http://journals.cambridge.org>>

## 2. ANTECEDENTES

### 2.1 HISTORIA DEL TSP<sup>12</sup>

Problemas matemáticos relacionados con el problema del viajante de comercio se trataron en el 1800 por el matemático irlandés Sir William Rowan Hamilton y por el matemático británico Thomas Penyngton Kirkman. La siguiente imagen es una fotografía de Juego Icosiano de Hamilton el cual consiste en dar con el recorrido de Hamilton pasando por las aristas de un dodecaedro para visitar una y sólo una vez, cada vértice, con la condición de que el de llegada coincida con el de partida. Este desafío se comercializó como un tablero con agujeros en cada nodo del grafo del dodecaedro y luego, fue distribuido en Europa en diversos formatos. Una buena discusión de los primeros trabajos de Hamilton y Kirkman se puede encontrar en el libro Teoría de Grafos 1736-1936 por NL Biggs, EK LLOYD, y RJ Wilson, Clarendon Press, Oxford, 1976.

**Ilustración 1. Juego Icosiano de Hamilton.**



**Fuente:** [http://www-groups.dcs.st-andrews.ac.uk/~history/Diagrams/Icosian\\_game.jpeg](http://www-groups.dcs.st-andrews.ac.uk/~history/Diagrams/Icosian_game.jpeg).

Los orígenes del TSP son Oscuros, la forma general de la TSP parece ser, se ha estudiado en secreto; por primera vez por los matemáticos a partir de la década de 1930 promovido por el matemático y economista Karl Menger en Viena y la Universidad de Harvard. El problema fue posteriormente promovido por Hassler Whitney y Merrill Flood reapareciendo en los círculos matemáticos en Princeton. En la década de 1940, fue estudiado por los estadísticos (Mahalanobis (1940), Jessen (1942), Gosh (1948), Marks (1948)) en relación con una solicitud agrícola, luego el matemático Merrill Flood lo popularizó entre sus colegas de la Corporación RAND. Finalmente, el TSP se hizo famoso como el prototipo de un problema difícil en la optimización combinatoria: “El examen de los tours de uno en uno es inadmisibles, debido a su gran número”.

Remítase al Anexo A, para la consulta del antecedente a cerca de la historia ilustrada en el avance para la solución del TSP.

---

<sup>12</sup> University of Waterloo. History of TSP. [On Line] URL: <<http://www.math.uwaterloo.ca/tsp/history/index.html>>. [Citado el 15 de Noviembre de 2013]

## **2.2 ANTECEDENTES DEL AG Y LA SOLUCIÓN APROXIMADA DEL TSP**

En la literatura existen autores cuyas investigaciones realizadas tienen como común objeto la solución aproximada del Problema del Agente Viajero (TSP) mediante la utilización del metaheurístico “Algoritmos Genéticos”. A continuación, se presenta una lista clasificada de algunos de los aportes consultados y una breve descripción de aquellos trabajos que de acuerdo a su enfoque, demuestran ser de gran relevancia en la guía de la temática que pretende la presente investigación de pregrado:

### **2.2.1 Clasificación de la Literatura Consultada.**

#### ***2.2.1.1 Estrategias para Evitar la Prematura Convergencia de la Población de Tratamiento.***

- ❖ Mechanisms to Avoid the Premature Convergence of Genetic Algorithms; por NICOARĂ, Elena Simona. [38]
- ❖ Reducing Premature Convergence Problem in Genetic Algorithm\_ Application on Travel Salesman Problem; por ZEYAD RAMADAN, Saleem. [44]
- ❖ Diversity-Guided Evolutionary Algorithms; por URSEM K., Rasmus. [43]

#### ***2.2.1.2 Modificación de los Operadores de Cruce como Estrategia de Diversificación de la Población de Tratamiento.***

- ❖ Developing Improved Greedy Crossover to Solve Symmetric Traveling Salesman Problem; por ISMKHAN, Hassan; ZAMANIFAR, Kamran. [27]
- ❖ Genetic Algorithm for the Traveling Salesman Problem Using Sequential Constructive Crossover Operator; por AHMED H. Zakir. [1]
- ❖ Very Greedy Crossover in a Genetic Algorithm for the Traveling Salesman Problem; por JULSTROM A. Bryant. [31]

#### ***2.2.1.3 Análisis de los Valores de los Parámetros de Entrada para un Buen Funcionamiento del Metaheurístico A.G.***

- ❖ Analysis of the Impact of Parameters Values on the Genetic Algorithm for TSP; por DIKA, Agni; MAXHUNI, Adnan; REXHEPI, Avni. [22]
- ❖ Optimal Mutation and Crossover Rates for a Genetic Algorithm Operating in a Dynamic Environment; por DAIDA M., Jason. STANHOPE A., Stephen. [15]
- ❖ Optimal Mutation Rates in Genetic Search; por BÄCK, Thomas. [6]

#### **2.2.1.4 Implementación de Variantes del AG para la Solución del TSP.**

- ❖ A Hybrid Genetic Algorithm - as New Approach to Solve Traveling Salesman Problem; por JAYALKSHMI, G. Andal; SATHI S. [29]
- ❖ An Efficient Hybrid Genetic Algorithm for the Traveling Salesman Problem; por KATAYAMA, Kengo; NARIHISHA, Hiroyuki. [33]
- ❖ A Hybrid Optimization Algorithm Based on Genetic Algorithm and Ant Colony Optimization; por PAPUTUNGAN, Irving Vitra; ZUKHRI, Zainudin. [40]

#### **2.2.1.5 Implementación de Metaheurísticos de Solución para la Resolución TSP.**

- ❖ Un Enfoque de Búsqueda TABÚ para el Problema del Agente Viajero; por GUERRERO BECERRA, John G.
- ❖ Traveling Salesman Problem Using Particle Swarm Optimization Algorithm; por CHEN, Wei LI, Wei; LIU, Hanmin; LUO, Wenjing; YAN. Xuesong; ZHANG, Can. [11]

### **2.2.2 Estrategias para Evitar la Prematura Convergencia de la Población de Tratamiento.**

**2.2.2.1 Mechanisms to Avoid the Premature Convergence of Genetic Algorithms; por NICOARÁ, Elena Simona. [38]** Las medidas para detectar la prematura convergencia son de hecho, medidas para el nivel de degeneración de la población, Srinivas y Patnaik (1994) utilizan como medida la diferencia entre la aptitud media de la población y la mejor aptitud en la población. En función de esta diferencia, de forma adaptativa, varían las probabilidades de cruce y mutación. Belea, R., Caraman S., Palade, V. en “*Diagnosing the Population State in a Genetic Algorithm Using Hamming Distance, the Proceedings of Knowledge-based and Intelligent Engineering Systems*” Proponen como medida estadística la media de la distancia hamming entre individuos y la varianza de las distancias Hamming, ambas independientes del número de genes y dimensión de la población. En este Paper, se proponen 2 mecanismos para mantener la diversidad de la población, los dos basados en el progreso medio de los operadores genéticos durante la evolución. Estos son: (1) La aplicación dinámica de los operadores de cruce y mutación y (2) Reinicialización Parcial de la población. El primero actúa lentamente, desde el inicio de la evolución hasta el final. Más precisamente, dos conjuntos de operadores (para la fase de cruce y para la fase de mutación) son usados en vez de dos operadores (1 operador cruce y 1 operador de mutación), en cada generación, 1 operador de cada conjunto es dinámicamente aplicado basado en la probabilidad de selección en función del progreso medio de la evolución. La Reinicialización parcial de la población es aplicada únicamente cuando el riesgo de prematura convergencia aparece. Fleming y Fonseca, para cada

generación inserta en la población un pequeño número de inmigrantes, generados aleatoriamente.

**2.2.2.2 Reducing Premature Convergence Problem in Genetic Algorithm Application on Travel Salesman Problem; por ZEYAD RAMADAN, Saleem. [44]** En el diseño del metaheurístico Algoritmo Genético (AG), se debe implementar medidas que aseguren la persistencia del proceso de evolución y la evasión a la convergencia prematura de la población. Por un lado, el uso apropiado de operadores de cruce y mutación ayudan a la creación de variación entre los individuos de la población reduciendo el inconveniente de la convergencia prematura; por otro lado, si la variación entre los individuos es muy alta, la información que es generada y transportada a través de las generaciones por medio de los individuos se perderá y así, una convergencia madura puede ser difícil de alcanzar. En este Paper para el buen diseño del AG se propone comprender la inducción de los focos de: “Estabilidad” mediante la utilización del operador de cruce “Frequency Crossover (FC)” y “Diversidad” a partir de 9 operadores diferentes de mutación agrupados en dos conjuntos, los cuales tienen respectivamente como objeto, la exploración y la explotación del espacio de soluciones. Se logra obtener resultados interesantes sobre las instancias tratadas en los experimentos, en donde se comprueba la relación dependiente entre estos dos focos para el buen desempeño del AG propuesto en la búsqueda de una solución asequible al problema del TSP.

### **2.2.3 Modificación de los Operadores de Cruce como Estrategia de Diversificación de la Población de Tratamiento.**

**2.2.3.1 Developing Improved Greedy Crossover to Solve Symmetric Traveling Salesman Problem; por ISMKHAN, Hassan; ZAMANIFAR, Kamran. [27]** En éste Paper los autores proponen la modificación del heurístico de cruce “Greedy Crossover (GX)” introducido por Greffenstette et al, el cual sólo puede ser utilizado mientras el sTSP sea resuelto por un Algoritmo Genético; aunque muchas otras investigaciones también centradas en tratar de generar una variante de mejor desempeño al GX, - manifiestan los autores del presente – todos resultan ser o más lentos o no tienen suficiente precisión. Producto de esta investigación es el operador de cruce “Improved Greedy Crossover (IGX)” el cual sustentado mediante los resultados experimentales obtenidos, demuestra un mejor rendimiento que los otros 6 con los que se comparó.

**2.2.3.2 Genetic Algorithm for the Traveling Salesman Problem Using Sequential Constructive Crossover Operator; por AHMED H. Zakir. [1]** En este Paper se propone el desarrollo de un nuevo operador de cruce, “Sequential Constructive Crossover (SCX)” para la generación de soluciones de alta calidad en el Problema del Agente Viajero mediante un Algoritmo Genético; la eficiencia del SCX es comparada contra otros dos operadores de cruce, considerando sus versiones originales: “Edge Recombination Crossover (ERX)” y “Generalized N-Point crossover (GNX)”, y según los resultados experimentales SCX demuestra ser un mejor operador de cruce en términos de calidad y tiempo de solución.

## **2.2.4 Análisis de los Valores de los Parámetros de Entrada para un Buen Funcionamiento del Metaheurístico A.G.**

**2.2.4.1 *Analysis of the Impact of Parameters Values on the Genetic Algorithm for TSP; por DIKA, Agni; MAXHUNI, Adnan; REXHEPI, Avni. [22]*** Muchos trabajos en el campo de la investigación de operaciones, han sido realizados con el ánimo de tratar de entender la mecánica de la interacción de los parámetros del Algoritmo Genético por medio de muchas técnicas, se entiende por parámetros del AG: el tamaño de la población de tratamiento, selección de los operadores genéticos del AG, probabilidades de los operadores, criterio de parada, etc. Por lo tanto, ¿Qué valores de los parámetros del AG, se deben usar para dar con una buena solución a un problema cualquiera? sigue siendo una pregunta abierta para los profesionales interesados en el tema. En este Paper, los autores prueban la relación entre (1) el tamaño de la población inicial, (2) probabilidad de mutación, y (3) número de generaciones para la corrida del AG propuesto, para la solución del TSP aplicado a la municipalidad de Kosovo.

**2.2.4.2 *Optimal Mutation and Crossover Rates for a Genetic Algorithm Operating in a Dynamic Environment; por DAIDA M., Jason. STANHOPE A., Stephen. [15]*** En este paper los autores concentran sus esfuerzos en la consecución de tasas de mutación y cruce que facilitan el desempeño del AG con base a un función de aptitud dinámica, resultando de esta investigación los siguientes dos productos: El primero, es una función de aptitud dinámica basada en el análisis previo de los escenarios estáticos y dinámicos planteados por la investigación, y el segundo y principal, es una relación general entre las tasas de mutación y cruce, útiles para una función de aptitud dinámica con una tasa específica de cambio en la “Distancia Hamming”, la cual podría posiblemente sugerir un vistazo de la utilidad que tendría un enfoque estándar del AG para la optimización de escenarios dinámicos. Dada la función de aptitud dinámica descrita en el paper, por medio del experimento diseñado, se obtuvo las estimaciones de rendimiento óptimo en dicha función de aptitud para diferentes tasas de cambio, mediante la implementación de un enfoque “Simple Meta-GA”, logrando obtener estimaciones de la configuración de los parámetros óptimos para las instancias de la Función de Ajuste de Aptitud Dinámico, cada uno con una tasa de cambio diferente.

### 3. JUSTIFICACIÓN

La situación que presenta la cotidianidad de las sociedades plantean un sinnúmero de problemas de soluciones expectantes, que pueden ser interpretados de manera correcta como oportunidades que enriquecen el conocimiento y la aplicación de metodologías que hacen posible la solución y por ende el mejoramiento continuo de la misma. Por tanto, es evidente experimentar en el medio, cierta presión por la generación de soluciones en un tiempo de costo relativamente razonable.

Este es el caso del Problema del Agente Viajero (*TSP*), de complejidad NP-Hard y amplia gama de posibilidades para su aplicación en la realidad, resulta sumamente improductivo el resolverse a través de un método de fuerza bruta, ya que el procedimiento presentaría una ineficiencia exponencial. Es por esto, que en la presente investigación se recurre a la implementación de un metaheurístico de búsqueda para su solución aproximada, contemplando su variante simétrica (*sTSP*) para efectos de comparación con la investigación hecha en la UPB-Bucaramanga “Un enfoque búsqueda tabú para el TSP”.

El Algoritmo Genético (GA), una rama principal de la Inteligencia Artificial propuesta por John H. Holland (1975) y fundamentado en el proceso de la evolución natural, es considerado como uno de los principales metaheurísticos para la resolución de problemas de gran dificultad, calificándole altamente para el fin de la presente investigación, por características como su naturaleza paralela en la búsqueda de una solución aproximada, utilización de operadores probabilísticos que garantizan en cierto grado la diversidad de la población estudio, facilidad para escapar de óptimos locales, y gran facilidad en la asimilación de su funcionamiento mediante el concepto de evolución, por mencionar algunos.

El desarrollo y producto que se obtenga de esta investigación pretende contribuir en el campo de la investigación de operaciones, a motivar el desarrollo tecnológico y experimental no solo al nivel UPB-Bucaramanga o AMB sino a nivel departamental, ya que es mínimo el número de investigaciones adelantadas en el terreno de la optimización en el que hagan uso de este tipo de técnicas especializadas para la resolución aproximada de problemas de alta complejidad.

“En la Actualidad el empleo de la Inteligencia Artificial crea el mundo en el que vivimos, por tanto no conocer estos métodos es una gran desventaja para cualquier ingeniero<sup>13</sup>”.

Dr. Pedro Ponce Cruz.

---

<sup>13</sup> PONCE CRUZ, Pedro. Inteligencia Artificial con aplicaciones a la ingeniería. 1 ed. México: Editorial Alfaomega grupo editor S.A., 2010. p. XIX. ISBN 978-607-7854-83-8 [Citado el 18 de Noviembre de 2013]

## 4. OBJETIVOS

### 4.1 OBJETIVO GENERAL

- ❖ Diseñar e implementar un Algoritmo Genético (AG) para dar solución aproximada al problema del agente Viajero en su variante simétrica (*sTSP*).

### 4.2 OBJETIVOS ESPECÍFICOS

- ❖ Identificar aspectos relevantes del Algoritmo Genético, considerando una corta revisión de la literatura, para resolver el problema *sTSP*.
- ❖ Diseñar e implementar el Algoritmo Genético en la resolución de un conjunto representativo de instancias, provistas por la biblioteca virtual de dominio público TSPLIB.
- ❖ Analizar y comparar los resultados obtenidos en la presente investigación, frente a los de la investigación “Un Enfoque de Búsqueda Tabú para el Problema del Agente Viajero” por el autor Jhon G. G. Becerra de la UPB - Bucaramanga – Colombia (Ts-TSP UPB) y la solución óptima hasta el momento suministrada por TSPLIB.
- ❖ Implementar el modelo de optimización lineal para el problema *sTSP* en cada una de las instancias TSPLIB consideradas, utilizando el software comercial GAMS, con el propósito de evaluar el desempeño computacional del Algoritmo Genético propuesto (AG-*sTSP*).

## 5. MARCO TEÓRICO

### 5.1 INVESTIGACIÓN DE OPERACIONES<sup>14</sup>

**5.1.1 Origen de la Investigación de Operaciones.** Las raíces de la investigación de operaciones se remontan a muchas décadas, cuando se hicieron los primeros intentos para emplear el método científico en la administración de una empresa. Sin embargo, el inicio de la actividad llamada *investigación de operaciones*, casi siempre se atribuye a los servicios prestados en los inicios de la Segunda Guerra Mundial. Debido a los esfuerzos bélicos, existía una necesidad urgente de asignar recursos escasos a las distintas operaciones militares y a las actividades dentro de la operación, en la forma más efectiva. Por esto las administraciones americana e inglesa hicieron un llamado a un gran número de científicos para que aplicaran el método científico para este y otros problemas estratégicos y tácticos. Con el desarrollo de métodos efectivos para el uso del nuevo radar, estos equipos de IO contribuyeron al triunfo del combate aéreo inglés. A través de sus investigaciones para mejorar el manejo de las operaciones antisubmarinas y de protección, jugaron también un papel, muy importante en la batalla del Atlántico Norte. Esfuerzos similares fueron de gran ayuda en la isla de Campaña del Pacífico.

Al terminar la guerra, el éxito de la investigación de operaciones en las actividades bélicas generó un gran interés en sus aplicaciones fuera del campo militar. Como la explosión de la industria seguía en curso, los problemas causados por el aumento de la complejidad y especializaciones dentro de las organizaciones pasaron de nuevo a primer plano. Comenzó a ser evidente para un gran número de personas incluso los consultores industriales que habían trabajado con o para los equipos de IO durante la guerra, que estos problemas eran básicamente los mismos que los enfrentados por la milicia, pero en un contexto diferente. Cuando comenzó la década del 1950, estos individuos habían introducido el uso de la investigación de operaciones en la industria, los negocios y el gobierno. Desde entonces esta disciplina se ha desarrollado con rapidez.

**5.1.2 Definición de Investigación de Operaciones.** Como su nombre lo dice, la investigación de operaciones significa "*hacer investigación sobre las operaciones*". Entonces la investigación de operaciones aplica a problemas que se refieren a la conducción y coordinación de operaciones (o actividades) dentro de una organización.

La parte de *investigación* en el nombre, significa que la investigación de operaciones usa un enfoque similar a la manera en la que se lleva a cabo la investigación en los campos científicos establecidos. Entonces en cierto modo, la IO incluye la investigación científica

---

<sup>14</sup> HILLIER S., Frederick; LIEBERMAN J., Gerald. Introducción a la Investigación de Operaciones. 4 Edición en español. Impreso en México D.F.: MCGRAW-HILL INTERAMERICANA EDITORES, S.A., 1997. 981 p. ISBN970-10-1022-1. p. 1-5 [Citado el 31 de Enero de 2014]

creativa de las propiedades fundamentales de las operaciones. Algunas otras características son:

- ❖ Su amplio punto de vista de manera organizacional, intenta resolver los conflictos de intereses entre las componentes de la organización de modo que el resultado sea el mejor para la organización completa.
- ❖ Intenta encontrar una mejor solución, (*llamada Solución Óptima*) para el problema bajo consideración. En lugar de contentarse con el estado de las cosas, la idea es identificar el mejor curso de acción posible.

**5.1.3 Impacto de la Investigación de Operaciones.** La Investigación de operaciones ha tenido un gran impacto en la mejora de la eficiencia de las organizaciones en todo el mundo. En el proceso, la investigación de operaciones ha hecho contribuciones significativas al incremento de la productividad dentro de la economía de varios países. Hay ahora más de 30 países que son miembros de la International Federation of Operational Research Societies (IFORS), en la que cada país cuenta con una sociedad de investigación de operaciones.

Remítase al Anexo B, Para la consulta de algunas aplicaciones reales de gran reconocimiento, en pro de estructurar una mejor idea en cuanto al amplio uso de la Investigación de Operaciones.

## **5.2 TEORÍA DE LA COMPLEJIDAD COMPUTACIONAL<sup>15</sup>**

**5.2.1 Definición de la Teoría de la Complejidad Computacional.** Es la parte de la teoría de la computación que estudia los recursos requeridos durante el cálculo para resolver un problema; un cálculo resulta complejo si es difícil de realizar. Los recursos comúnmente estudiados son el tiempo (número de pasos de ejecución de un algoritmo para resolver un problema) y el espacio (cantidad de memoria utilizada para resolver un problema). A estos recursos se les puede añadir otros, tal como el número de procesadores necesarios para resolver el problema en paralelo, etc.

Si un cálculo requiere más tiempo que otro, decimos que es más complejo y se le caracteriza por complejidad temporal; pero si un cálculo requiere más espacio de almacenamiento que otro, en este caso hablamos de una complejidad espacial.

Aunque actualmente la mayoría de los algoritmos resueltos por las máquinas tienen como máximo una complejidad o costo computacional polinómico, es decir, la relación entre el tamaño del problema y su tiempo de ejecución es polinómica – a estos problemas se les

---

<sup>15</sup> CORTÉZ, Augusto. Ensayo: Teoría de la complejidad computacional y teoría de la computabilidad. [En Línea] Disponible en Pdf. URL: <[http://sisbib.unmsm.edu.pe/bibvirtualdata/publicaciones/risi/n1\\_2004/a14.pdf](http://sisbib.unmsm.edu.pe/bibvirtualdata/publicaciones/risi/n1_2004/a14.pdf)> Revista de Investigación de Sistemas e Informática, 2004, ISSN: 1815-0268 (impreso) [Citado el 20 de Noviembre de 2013]

clasifica en la clase **P**-, por otro lado existen problemas con costo no polinomial a los cuales se les agrupa en la clase **NP** y cuentan con la particularidad de que estos problemas no tienen una solución de costo algorítmico, es decir, una máquina no puede resolverlos en un tiempo razonable.

Existe una escala para medir la complejidad, la que incluye, entre otros:

- ❖ **P**, Resoluble en tiempo polinómico.
- ❖ **P-Completo (P-Complete, P-Hard)**, los problemas más difíciles en P.
- ❖ **NP**, problemas con respuestas positivas verificables en tiempo polinómico y
- ❖ **NP-Completo (NP-Complete, NP-Hard)**, los más difíciles problemas de NP.

El interés de Turing acerca de la capacidad de las máquinas para pensar lo llevó a desempeñar un papel importante en el desarrollo de las computadoras, no solo teóricas sino reales. Así, se refuerza la tesis de Church-Turing que dice que si una máquina de Turing no puede resolver un problema, entonces ningún computador puede hacerlo, ya que no existe algoritmo para obtener una solución. Por tanto, las limitaciones corresponden a procesos computacionales y no a la tecnología.

**5.2.2 Clasificación NP-Hard del TSP.**<sup>16</sup> Richard M. Karp demostró en 1972, que el problema de ciclo Hamiltoniano era de naturaleza NP-complete, lo cual implica la clasificación NP-difícil del problema del agente viajero (TSP). Esto, mediante una explicación matemática por la aparente dificultad computacional de encontrar recorridos óptimos para el mismo.

### 5.3 PROBLEMA DEL AGENTE VIAJERO

**5.3.1 Definición del Problema del Agente Viajero Simétrico (sTSP).**<sup>17</sup> Dada un conjunto las ciudades y costos de viaje entre cada par de ellas, el problema del agente viajero (TSP) se puede definir como: Tratar de encontrar la forma más barata de visitar todas las ciudades y regresar a su punto de partida. Teniendo en cuenta, que los costos de viaje son simétricos en el sentido de que viajando de la ciudad X a la ciudad Y cuesta tanto como viajar de Y a X (Entendiéndose costo como distancia, valor monetario, tiempo o cualquier otra variable que guarde proporcionalidad con el recorrido).

**5.3.2 Complejidad.**<sup>18</sup> Dado que n es el número de ciudades que se visitarán en el viaje del agente, Es fácil calcular el número de diferentes recorridos para n ciudades:

---

<sup>16</sup> KARP, R. (1972). Reducibility among combinatorial problems: In Complexity of Computer Computations. [En Línea] Disponible en Pdf. URL: <<http://cgi.di.uoa.gr/~sgk/teaching/grad/handouts/karp.pdf>> Plenum Press, pp. 85-103. New York. [Citado el 20 de Noviembre de 2013]

<sup>17</sup> University of Waterloo. The Problem of TSP. [En Línea] URL: <<http://www.math.uwaterloo.ca/tsp/problem/index.html>> [Citado el 22 de Noviembre de 2013]

<sup>18</sup> University of Waterloo. Number of Tours. [En Línea] URL: <<http://www.math.uwaterloo.ca/tsp/problem/pcb3cnt.html>> [Citado el 22 de Noviembre de 2013]

- ❖ El número de ciclos posibles puede calcularse de manera sencilla: El origen viene determinado, por lo que restarían  $(n - 1)$  puntos para empezar, a continuación se elige cualquiera de los  $(n - 2)$  restantes y así sucesivamente, se multiplican de la forma  $n-1 \times n-2 \times n-3 \times \dots \times 3 \times 2 \times 1$ . Ahora, ya que los costos de viaje no dependen de la dirección que se tome para su recorrido.

$$\text{Number of Possible Solutions (Routes) for sTSP} = \frac{(n - 1)!}{2}$$

**5.3.3 Formulación Matemática del TSP.** Dado que en la literatura existen varias formulaciones matemáticas que describen la naturaleza del TSP simétrico, para efectos de su comprensión y desarrollo de la presente investigación aplicada de pregrado, se utilizará la formulación propuesta George Dantzing, Ray Fulkerson y Selmer Johnson<sup>19</sup> donde según la teoría de grafos  $G = (V, E)$ ;  $c_{ij} = c_{ji}$  es la matriz de coste simétrica de viajar de  $i$  a  $j$ ,  $x_{ij}$  es la variable dicotómica de decisión para todo  $(i, j)$ ,  $V$  el conjunto de  $n$  vértices  $\{1, 2, \dots, n\}$  y  $E$  es el conjunto de arcos no dirigidos  $(i, j)$  para el caso simétrico del Agente Viajero (sTSP):

$$\text{Minimizar: } \sum_{i < j} c_{ij} x_{ij} \quad (1),$$

$$\text{Sujeto a: } \sum_i x_{ij} = 1, \sum_j x_{ij} = 1; \quad \forall (i, j) \in V \quad (2),$$

$$x_{ij} \in \{0, 1\}; \quad \forall (i, j) \in E \quad (3),$$

$$\sum_{i > j = k} x_{ik} + \sum_{j < i = k} x_{kj} = 2; \quad (k \in V) \quad (4),$$

$$\sum_{i, j \in S} x_{ij} \leq |S| - 1; \quad \forall (i, j) \in S, \quad (S \subset V, 2 \leq |S| \leq n - 1) \quad (5).$$

(1), es la función objetivo de costo a minimizar,

(2), son las restricciones que limitan el uso de arcos a 1 sola vez, es decir, que la llegada y la partida a cada uno los mismos, ocurra una sola vez,

(3), valida la variable de decisión  $x_{ij}$  como una variable binaria al adjudicarle los valores de 0 y 1 según aplique el caso determinado por las restricciones en (2),

(4), asegura que cada segmento de ruta esté conformado por 2 arcos y

(5), Evita que se cumplan (2), (3), y (4) en presencia de 1 o más subciclos inconexos, obligando así a que la solución que se presente, describa una ruta única que recorra en su totalidad el grafo.

---

<sup>19</sup> G. Dantzig, R. Fulkerson, S. Johnson. Solution of a Large-Scale Traveling-Salesman Problem [En Línea] URI: <<http://www.iro.umontreal.ca/~gendron/IFT6551/LECTURES/TSP.pdf>> [Citado el 22 de Noviembre de 2013]

### 5.3.4 Variantes del TSP.

Tabla 1. Variantes del Problema del Agente Viajero (TSP).

DEL INGLÉS	ESPAÑOL
Symmetric Traveling Salesman Problem (sTSP)	Problema del Agente Viajero Simétrico
Asymmetric Traveling Salesman Problem (aTSP)	Problema del Agente Viajero Asimétrico
Hamiltonian Cycle Problem (HCP)	Problema de Ciclos Hamiltonianos
Sequential ordering Problem (SOP)	Problema de Ordenamiento de Secuenciación
Capacited Vehicle Routing Problem (CVRP)	Problema de Ruteo de Vehículos con Restricciones de Capacidad
TSP with Distances 1 and 2, TSP(1,2)	TSP con Distancias 1 y 2; TSP(1,2)
Prize-Collecting Traveling Salesman Problem (PCTSP)	TSP-Recolección de Premios
The Euclidean Traveling Salesman Problem	Problema de Selección del Agente Viajero Euclidiano
Circulant Traveling Salesman Problem	Problema del Agente Viajero Circulante
Remote TSP	TSP Remoto
Max TSP	TSP Máximo
On-Line Traveling Salesman Problem	Problema del Agente Viajero en Línea

Fuente: Elaborado por el Autor.

## 5.4 METAHEURÍSTICOS<sup>20</sup>

**5.4.1 Definición de Metaheurísticos.** Este término deriva de la palabra griega “Heuriskein” que significa encontrar o descubrir y se usa en el ámbito de la optimización para describir una clase de algoritmos de resolución de problemas; algunas clases de problemas de optimización son relativamente fáciles de resolver; sin embargo, muchos otros tipos de problemas de optimización son muy difíciles de resolver, entendiéndose así el término científico *NP-Hard*, en términos coloquiales, como un problema de optimización difícil al que no se le puede garantizar, encontrar la mejor solución posible en un tiempo razonable.

La existencia de una gran cantidad y variedad de problemas difíciles, que aparecen en la práctica y que necesitan ser resueltos de forma eficiente, es la razón que ha impulsado el

---

<sup>20</sup> MARTÍ, Rafa. Procedimientos Metaheurísticos en Optimización Combinatoria. [En Línea] Disponible en Pdf: <<http://www.uv.es/rmartipaper/docs/heur1.pdf>> Universidad de Valencia, Facultad de matemáticas, departamento de estadística e investigación operativa. p. 1-2 [Citado el 23 de Noviembre de 2013]

desarrollo de procedimientos eficientes para encontrar buenas soluciones aunque no fueran óptimas, es decir, aproximados. A estos métodos, en los que la rapidez del proceso es tan importante como la calidad de la solución obtenida, se les denominan heurísticos o aproximados.

En Díaz y otros (1996) se recogen hasta ocho definiciones diferentes de algoritmo heurístico, entre las que se destaca la siguiente:<sup>21</sup>

*“Un método heurístico es un procedimiento para resolver un problema de optimización bien definido mediante una aproximación intuitiva, en la que la estructura del problema se utiliza de forma inteligente para obtener una buena solución”.*

Existen otras razones para utilizar métodos heurísticos, entre las cuales se destacan:

- ❖ El problema es de una naturaleza tal que no se conoce ningún método exacto para su resolución.
- ❖ Aunque existe un método exacto para resolver el problema, su uso es computacionalmente muy costoso.
- ❖ El método heurístico es más flexible que un método exacto, permitiendo, por ejemplo, la incorporación de condiciones de difícil modelización.
- ❖ El método heurístico se utiliza como parte de un procedimiento global que garantiza el óptimo de un problema.

**5.4.2 Clasificación de Metaheurísticos de Búsqueda.** En la Tabla 2. Se puede observar una clasificación de los metaheurísticos de búsqueda empleados para la resolución de problemas combinatorios.<sup>22</sup>

**Tabla 2. Clasificación de Metaheurísticos de Búsqueda.**

<b>METAHEURÍSTICOS DE BÚSQUEDA</b>	
<b>De Búsquedas Locales</b>	<b>De Búsquedas Globales</b>
• Búsquedas no informadas	• Búsqueda multi-arranque

<sup>21</sup> DÍAZ, A., GLOVER, F., GHAZIRI, H.M., GONZALEZ, J.L., LAGUNA, M, MOSCATO, P. y TSENG, F.T. (1996), Optimización Heurística y Redes Neuronales, Paraninfo, Madrid. [Citado el 23 de Noviembre de 2013]

<sup>22</sup> BRITO SANTANA, Julio; CAMPOS RODRÍGUEZ, Clara; GARCÍA LÓPEZ C. Félix; GARCÍA TORRES, Miguel; MELIÁN BATISTA, Belén; MORENO P´EREZ, José A.; MORENO VEGA, J. Marcos; Metaheurísticas: una revisión actualizada. [En Línea] Disponible en Pdf. URL: <<http://jamoreno.webs.ull.es/www/papers/paper37.pdf>> Grupo de Computación Inteligente Universidad de La Laguna. Departamento de Estadística, Investigación Operativa y Computación. Documento de Trabajo no 02/2004. [Citado el 24 de noviembre de 2013]

**Tabla 2. Clasificación de Metaheurísticos de Búsqueda. (Continuación).**

<ul style="list-style-type: none"> <li>• Búsquedas no informadas</li> <li>• Búsquedas locales monótonas</li> </ul>	<ul style="list-style-type: none"> <li>• Búsqueda multi-arranque</li> <li>• Búsqueda por entornos variables</li> <li>• Búsquedas no monótonas</li> </ul>
<b>De Búsquedas Basadas en Poblaciones o Evolutivas</b>	<b>Metaheurísticos de Búsqueda</b>
<ul style="list-style-type: none"> <li>• Algoritmos genéticos</li> <li>• Algoritmos de estimación de distribuciones             <ul style="list-style-type: none"> <li>• Búsqueda dispersa</li> </ul> </li> <li>• Algoritmos bionómicos</li> <li>• Búsqueda por reencadenamiento de camino</li> </ul>	<ul style="list-style-type: none"> <li>• Optimización de partículas inteligentes             <ul style="list-style-type: none"> <li>• Búsqueda por ruido</li> <li>• Búsqueda local iterada</li> <li>• Heurística de concentración</li> </ul> </li> <li>• Optimización de partículas inteligentes             <ul style="list-style-type: none"> <li>• Búsqueda local guiada</li> <li>• Metaheurístico FANS</li> <li>• Búsqueda reactiva</li> <li>• Algoritmos Meméticos</li> </ul> </li> </ul>
<b>Otros Metaheurísticos</b>	
<ul style="list-style-type: none"> <li>• Redes neuronales</li> <li>• Colonias de hormigas</li> <li>• Optimización extrema</li> <li>• Metaheurísticos GRASP</li> </ul>	<ul style="list-style-type: none"> <li>• Satisfacción de restricciones</li> <li>• Metaheurísticos multiobjetivo</li> <li>• Metaheurísticos paralelas             <ul style="list-style-type: none"> <li>• Hiperheurísticos</li> </ul> </li> </ul>

**Fuente: Elaborado por el Autor.**

**5.4.3 Métodos Heurísticos.**<sup>23</sup> Existen muchos métodos heurísticos de naturaleza muy diferente, por lo que es complicado dar una clasificación completa. Además, muchos de ellos han sido diseñados para un problema específico sin posibilidad de generalización o aplicación a otros problemas similares; el siguiente esquema trata de dar unas categorías amplias, no excluyentes, en donde ubicar a los heurísticos más conocidos. [Remítase a la nota al pie 20].

## **5.5 ALGORITMOS GENÉTICOS**

**5.5.1 Definición de los Algoritmos Genéticos.** Propuesto por John H. Holland (1975) y fundamentado en el proceso de la evolución natural, los algoritmos genéticos (GA) son un método de búsqueda dirigida basado en probabilidad (estocástico), que siguen una metáfora biológica: "Los AG ven la resolución de un problema de optimización como una competencia entre una población, donde las soluciones candidatas al problema, evolucionan de tal forma son evaluadas por una función de "aptitud" (Fitness Function) que decide probabilísticamente si contribuirá o no, a la próxima generación de soluciones.

<sup>23</sup> MARTI. Op. Cit., p. 4.

Entonces, a través de operaciones análogas a la transferencia de genes en la reproducción sexual (operadores genéticos), el algoritmo crea una nueva población de soluciones candidatas”.

Al comienzo de la corrida de un algoritmo genético, la población de individuos (cadena de valores o cromosomas) es generada de forma completamente aleatoria y la decodificación de cada uno de estos, representa una solución posible al problema en cuestión.<sup>24</sup>

**5.5.2 Operadores Genéticos de los AGs.**<sup>25</sup> Partiendo de la premisa de emplear la evolución natural como un procedimiento de optimización, los AGs se caracterizan por tener operadores básicos que hacen posible la transición de una generación a otra:

**5.5.2.1 Selección.** Mecanismo de selección individual (cadena) para la reproducción acorde con la *función de aptitud* (valor de la función objetivo). Los algoritmos de selección serán los encargados de escoger qué individuos van a disponer de oportunidades de obtener copias y reproducirse y cuáles no.

**5.5.2.2 Cruzamiento.** Es el principal operador genético que provee un mecanismo para heredar características a su descendencia; es un método de fusión sobre la información genética de dos individuos.

**5.5.2.3 Mutación.** En la evolución real, el material genético puede ser alterado en forma aleatoria debido a un error en la reproducción o la deformación de genes. En los AG, la mutación se realiza como una deformación aleatoria de las cadenas, produciendo cambios incrementales al azar en la descendencia, efectuando cambios aleatorios en los valores del alelo en algunos genes.

**5.5.3 Pseudocódigo General del Algoritmo Genético.**<sup>26</sup> **Tc** es la tasa de cruce o el parámetro que determina la tasa en la que dicho operador es aplicado, **Tm** es el equivalente para la tasa de mutación, **Tp** tamaño de la población (# de cromosomas) y **MaxG** puede entenderse como el criterio de parada del algoritmo.

#### Ilustración 2. Pseudocódigo del AG.

```
Algorithm: Genetic Algorithm
1:  Procedure GENETIC (Tc, Tm, Tp, MaxG)
2:    Pop ← Generate Population (Tp)
3:    Pop ← Evaluation (Pop)
4:    For i = 1....MaxG do
5:      Pop ← Selection (Pop)
```

<sup>24</sup> DE LIMA JUNIOR, Francisco C.; DORIA NETO, Adrião D.; DE MELO, Jorge Dantas. Hybrid Metaheuristics Using Reinforcement Learning Applied to Salesman Traveling Problem. Traveling Salesman Problem, Theory and Applications. December, 2010. p. 213-236. ISBN 978-953-307-426-9. [Citado el 25 de noviembre de 2013]

<sup>25</sup> PONCE CRUZ, Pedro. Inteligencia Artificial con aplicaciones a la ingeniería. 1 ed. México: Editorial Alfaomega grupo editor S.A., 2010. p. 291. ISBN 978-607-7854-83-8 [Citado el 25 de Noviembre de 2013].

<sup>26</sup> Op. Cit., p. 217.

**Ilustración 2. Pseudocódigo del AG. (Continuación).**

```

6:          With probability  $T_c$  do:
               $Pop \leftarrow Crossover (Pop)$ 
7:          With probability  $T_m$  do:
               $Pop \leftarrow Mutation (Pop)$ 
8:          End For
9:          Return the best solution in  $Pop$ 
10: End Procedure

```

---

**Fuente: Hybrid Metaheuristics Using Reinforcement Learning Applied to Salesman Traveling Problem.**

**5.4.1 Aplicaciones Recientes de los Algoritmos Genéticos.** Remítase al Anexo C, para la consulta de algunas aplicaciones de los algoritmos genéticos, referencias actuales en la ciencia, ingeniería, los negocios y ciencias sociales.

**5.5.5 Estructura Equivalente entre la Genética y el AG.**<sup>27</sup> La siguiente tabla, contiene diferentes expresiones utilizadas en la genética y su estructura equivalente al AG:

**Tabla 3. Similitud Conceptual entre la Genética y AGs.**

EVOLUCIÓN NATURAL	ALGORITMO GENÉTICO
Genotipo	Código de Cadena
Fenotipo	Punto Sin codificar
Cromosoma	Cadena
Gen	Posición de cadena
Alelo	Valor de una posición determinada
Función de aptitud o aptitud	Valor de la función objetivo

**Fuente: Inteligencia Artificial con aplicaciones a la ingeniería.**

**5.5.5.1 Genotipo.** Expresión genética de un organismo o estructura genética del organismo. La información contenida en el genoma.

**5.5.5.2 Fenotipo.** Características físicas de un organismo, atribuibles a la expresión de su fenotipo.

**5.5.5.3 Cromosoma.** Es la molécula única de ADN. Cadena de valores que contiene la información atribuida al genotipo.

---

<sup>27</sup> Op. Cit., p.13.

**5.5.5.4 Gen.** Especifica la herencia de un carácter que realizan diferentes funciones en la cadena.

**5.5.5.5 Alelo.** El valor de un gen. Determina el carácter controlado por el gen.

**5.5.5.6 Función de aptitud.** Es un tipo especial de función que cuantifica la optimalidad de una solución. Se traduce en un cromosoma óptimo para que sus bases sean combinadas con cualquier otra técnica para la producción de una nueva generación que sea mejor a las anteriores.

## 5.6 HEURÍSTICO DE AHORROS DE CLARKE AND WRIGHT<sup>28</sup>

Propuesto inicialmente para problemas de rutas de vehículos, el algoritmo se basa en combinar sucesivamente subtours hasta obtener un ciclo Hamiltoniano. Los subtours considerados tienen un vértice común llamado *base o punto de partida*.

El procedimiento de unión de subtours se basa en eliminar las aristas que conectan dos vértices de diferentes subtours con el vértice base, uniendo posteriormente los vértices entre sí. Se le llama *ahorro* a la diferencia del coste entre las aristas eliminadas y la añadida.

### 5.6.1 Procedimiento para la Obtención del Tour Factible Clarke & Wright.

- (1) El punto 0 corresponde al punto de partida.
- (2) Se calcula el ahorro para todas las parejas de clientes. El ahorro de la pareja  $i, j$  está definido por la fórmula:

$$S_{(i,j)} = D_{(i,0)} + D_{(0,j)} - D_{(i,j)}$$

- (3) Se ordenan los ahorros en forma descendente.
- (4) Usando los elementos de la lista anterior:
  - (4.1) Si al unirse dos clientes por un tramo de la ruta, ésta resulta ser factible y respeta todas las restricciones del problema, entonces se agrega este tramo a la solución. Si no, se rechaza. Para una ruta factible, se eliminan todos los tramos no analizados que conectan con ese tramo.

---

<sup>28</sup> PINILLA GARCÍA, Vicente. Investigación Operacional: Introducción teórica y aplicaciones en administración. 1 ed. Colombia, Bogotá D.C.: Ediciones Uniandes, 2005. p.140. ISBN958-695-161-8 [Citado el 24 de Enero de 2014]

- (4.2) Se toma la siguiente pareja de la lista del paso (3) y se repite el paso (4.1) se continúa con este paso (4.2) hasta terminar de analizar todos los elementos de la lista del paso (3).
- (5) Se unen todos los tramos en ese orden  $(0,j), (j,k), (k,p), \dots, (m,0)$  a fin de formar todas las rutas posibles.

### 5.6.2 Algoritmo de Ahorros Clarke & Wright.<sup>29</sup>

#### Ilustración 3. Algoritmo de Ahorros C & W.

##### Algoritmo de ahorros C & W

###### Inicialización

Tomar un vértice  $z \in V$  como base.  
Establecer los  $n-1$  subtours  $[(z, v), (v, z)], \forall v \in V \setminus \{z\}$ .

###### Mientras (queden dos o más subtours)

Para cada par de subtours calcular el ahorro de unirlos al eliminar en cada uno, una de las aristas que lo une con  $z$  y conectar los vértices asociados.  
Unir los dos subtours que produzcan un ahorro mayor.

Fuente: Procedimientos Metaheurísticos en Optimización Combinatoria.

<sup>29</sup> MARTÍ, Rafael. Procedimientos Metaheurísticos en Optimización Combinatoria. [En Línea] Disponible en Pdf: <<http://www.uv.es/rmarti/paper/docs/heur1.pdf>> Universidad de Valencia, Facultad de matemáticas, departamento de estadística e investigación operativa. p. 22. [Citado el 24 de Enero de 2014]

## 6. DISEÑO METODOLÓGICO

Para este problema de investigación aplicada de pregrado -Problema del Agente Viajero- el método científico propone una metodología acertada para su abordaje, guía y desarrollo en pro de la consecución de los resultados proyectados como aportes sustanciales en el campo de la investigación de operaciones.

Lo anterior infiere que el investigador intervenga, en forma directa, en cada una de las etapas del método; permitiendo a través del experimento(s) la ampliación de la concepción sobre la naturaleza del fenómeno estudiado, a una más cercana.

Las conclusiones que se obtengan pueden ser interpretadas como base científica para el desarrollo de futuras investigaciones, de modo que el conocimiento generado sea cada vez más especializado. A continuación se expone las etapas del método a manera de proceso de mejora continua, ajustadas a las necesidades de la presente investigación.

**Ilustración 4. Método Científico.**



Fuente: Elaborado por el Autor.

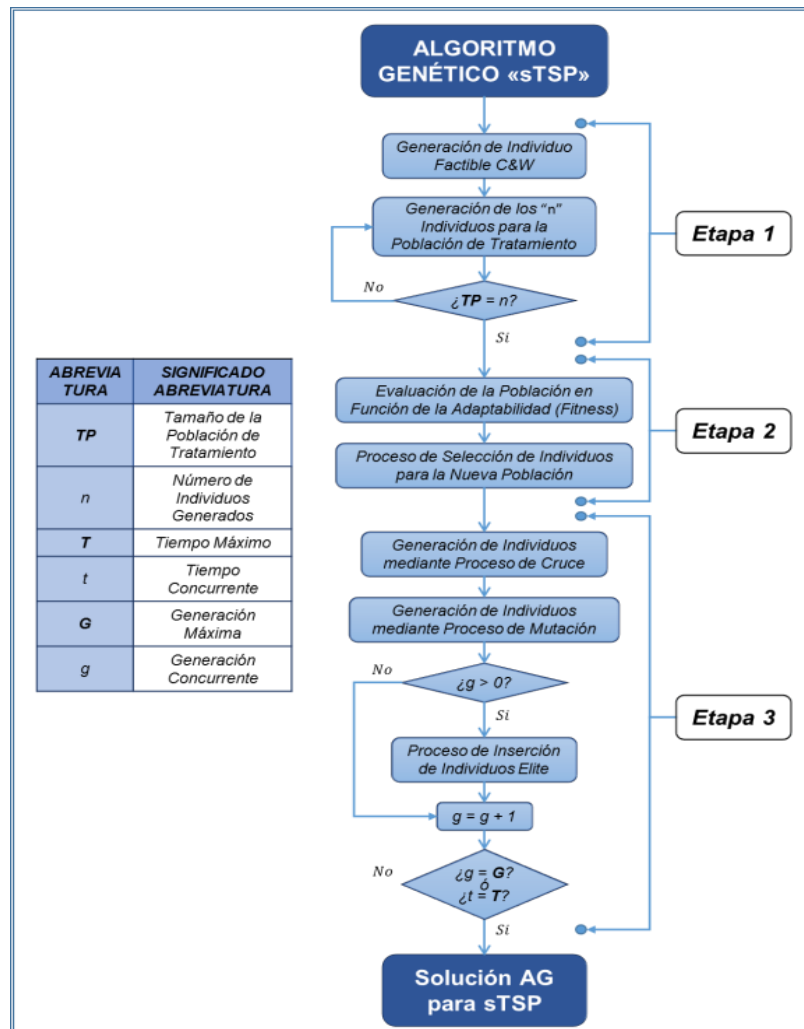
## 7. UN ALGORITMO EVOLUTIVO PARA EL PROBLEMA DEL AGENTE VIAJERO

### 7.1 ASPECTOS RELEVANTES Y ANÁLISIS DEL PSEUDOCÓDIGO DEL AG PARA SU IMPLEMENTACIÓN EN LA SOLUCIÓN APRÓX. DEL sTSP

Seguidamente se esboza a manera de guía, la aplicación de la metodología anteriormente definida mediante la identificación y análisis de aspectos de gran relevancia de cada una de las etapas que el AG comprende para su diseño, desarrollo e implementación; para efectos de la presente investigación: “el problema del Agente Viajero en su variante simétrica (sTSP)”.

#### 7.1.1 Pseudocódigo General del AG Propuesto (AG-sTSP).

Ilustración 5. Pseudocódigo General del AG Propuesto (AG-sTSP).



ABREVIATURA	SIGNIFICADO ABREVIATURA
TP	Tamaño de la Población de Tratamiento
n	Número de Individuos Generados
T	Tiempo Máximo
t	Tiempo Concurrente
G	Generación Máxima
g	Generación Concurrente

Fuente: Elaborado por el Autor.

## 7.1.2 Etapa 1: Generación de la Población de Tratamiento.

**7.1.2.1 Representación, Codificación de Individuos**<sup>30</sup>. Los problemas de permutación requieren un arreglo optimal de un conjunto de símbolos en una lista. El TSP es un problema tal, en donde un símbolo puede ser usado para representar una ciudad y el arreglo ordenado de tales símbolos en una lista, pueden describir la secuencia en la que el vendedor o agente viajero visitaría cada ciudad, formando así un circuito a recorrer.

**Para Experimentación.** La representación codificada de cada individuo (cromosoma, ruta, solución posible, etc.) de la población (Región del espacio de búsqueda), se hará mediante el arreglo ordenado (sin redundancia o repetición) de una lista de valores enteros de longitud  $l+2$ , donde  $l$ , corresponde al número total de ciudades (sin contar el depósito) que contempla la instancia a tratar por el AG. Ej.:

### Ilustración 6. Representación, Codificación del individuo para TSP.

0	1	2	3	...	...	...	$l-2$	$l-1$	$l$	0
---	---	---	---	-----	-----	-----	-------	-------	-----	---

Fuente: Elaborado por el Autor.

**7.1.2.2 Tamaño de la Población de Tratamiento.** Desde la perspectiva de evolución genética, un tamaño de población pequeña haría susceptible en mayor grado a la metaheurística al efecto de una convergencia prematura, por razón de que no se tendría suficiente diversidad para evitar o mitigar este impacto; por otro lado, una población muy grande incrementaría exponencialmente el tiempo de corrida del AG haciéndolo ineficiente para su uso (BASNET 2012).

**Para Experimentación.** Se toma en cuenta la sugerencia para la generación del tamaño de la población tratamiento hecha por J. T. Alander, el cual, basándose en evidencia empírica indica que un tamaño de población comprendida entre  $l$  y  $2l$  es suficiente para atacar con éxito los problemas por el considerados (sTSP)<sup>31</sup>. Para efectos de funcionalidad y desempeño del AG propuesto (AG-sTSP), se desarrollará un diseño de experimentos que permita establecer la magnitud de este parámetro de entrada en relación a la instancia a procesar.

**7.1.2.3 Generación de la Población Inicial de Tratamiento.** Los AGs inician la búsqueda a partir de una población de soluciones candidatas, la cual comúnmente suele ser concebida de forma completamente aleatoria. Por otro lado, también se podrían tomar como población inicial aquellos individuos que surgen como producto de algunos otros

<sup>30</sup> DASGUPTA, Dipankar; MICHALEWICZ, Zbigniew. (Eds). Evolutionary Algorithms in Engineering Applications. Alemania: Editorial Springer-Verlag Berlín Heidelberg, 1997. p. 33. ISBN 3-540-62021-4 [Citado el 6 de Diciembre de 2013].

<sup>31</sup> J.T. Alander (1992). On optimal Population Size of Genetic Algorithms. Proceedings CompEuro - Computer Systems and Software Engineering, 6th Annual European Computer Conference. May, 1992. p. 65-70. ISBN 0-8186-2760-3 [Citado el 7 de Diciembre de 2013].

heurísticos de resolución empleados también, en la generación de buenas soluciones iniciales aproximadas para el problema tomado en cuenta<sup>32</sup>.

**Para Experimentación.** Teniendo en cuenta que las soluciones iniciales buenas, usualmente producen buenos resultados con menor costo de tiempo<sup>33</sup>; mediante la heurística de ahorro propuesta por Clarke & Wright, se hará la inserción de un individuo semilla (de calidad aparentemente buena) con el propósito de que a partir de este, bajo un proceso de alteración probabilística “**Bit-Split**”, se constituya una población de individuos de caracteres diversificados.

### 7.1.3 Etapa 2: Evaluación de la Población de Tratamiento.

**7.1.3.1 Convergencia Prematura del AG.** “La optimización por medio de algoritmos genéticos a menudo viene junto con el sesgo de convergencia prematura, especialmente en los problemas multimodales o difíciles” (NICOARĂ, 2009); esta se puede definir como la reducción progresiva de la capacidad del algoritmo para encontrar soluciones buenas en el espacio de búsqueda y se presenta cuando los genes de los individuos mejor adaptados (de mayor valor de aptitud), dominan rápidamente a los genes de una población.

**7.1.3.2 Definición de la Función Aptitud<sup>34</sup>.** La función de aptitud o del inglés “Fitness Function” es el método que cuantifica la “bondad” del cromosoma (Individuo), calificando su grado de idoneidad para posteriores tratamientos (cruce y mutación) en pos de su participación en poblaciones de las siguientes generaciones.

**Para Experimentación.** El AG fue diseñado para la resolución de problemas de maximización; así, la selección proporcional considera como cromosomas de mejor calidad a aquellas que poseen un mayor valor de aptitud. Para este caso, en un problema de minimización como estrategia equivalente al esquema de selección proporcional de un problema de maximización, la literatura comúnmente sugiere la “transformación” de la función objetivo en la siguiente:

$$F(x) = \frac{1}{f(x)} \quad (1)$$

Donde  $f(x)$  calcula el costo (valor) del tour representado por el cromosoma.

---

<sup>32</sup> Universidad de la República de Uruguay Facultad de ingeniería. Material de enseñanza: Técnicas de Computación Evolutiva. [En Línea] Disponible en Pdf. URL: <<http://www.fing.edu.uy/inco/cursos/geneticos/ae/CE.pdf>> [Citado el 7 de Diciembre de 2013].

<sup>33</sup> NASEEM J, Sadaf; YANG, Shengxiang. A Hybrid Genetic Algorithm and Tabu Search Approach for Post Enrolment Course Timetabling. Journal of Scheduling; December, 2011, Vol. 14 Issue 6, p. 617-637. ISSN: 10946136. [Citado el 7 de Diciembre de 2013].

<sup>34</sup> AHMED, Zakir H. Genetic Algorithm for the Traveling Salesman Problem using Sequential Constructive Crossover Operator. International Journal of Biometrics & Bioinformatics (IJBB). January, 2010. Vol. 3, Issue 6, p. 96-105. p. 98. ISSN: 1985-2347. [Citado el 7 de Diciembre de 2013].

**7.1.3.3 Proceso de Selección Natural**<sup>35</sup>. Este proceso consta de una dinámica de oportunidad que prioriza la supervivencia y participación en los procesos de transformación de cadenas (cruce y mutación) de aquellos individuos que presentan una mejor configuración en una población dada; dicho proceso viene condicionado habitualmente mediante probabilidades según los valores de aptitud o *Fitness*, guiando el desarrollo del AG hacia mejores soluciones.

**Para Experimentación.** Se hará uso del *Método de Selección Estocástico Proporcional* o también llamado *Método de la Ruleta*, favoreciendo en la elección (supervivencia) para su participación en los procesos de cruce y mutación, no solo a aquellos individuos que presentan los mayores valores de aptitud en la población de tratamiento, sino también, a aquellos que de baja adaptabilidad presentan una probabilidad menor de selección.

#### **7.1.4 Etapa 3: Evolución de la Población de Tratamiento.**

Es aquí, donde dada la condición: de que “después de haber finalizado la presente etapa y el criterio de parada no haya sido sufragado”; El AG-sTSP, entra en un ciclo temporal de exploración que tiene como objeto el procesamiento de la nueva información (nueva población de tratamiento) para la búsqueda una posible mejor solución al sTSP.

**7.1.4.1 Criterio de Parada.** “*Stopping Criterion*” es un parámetro de entrada al problema, el cual describe de forma puntual, la(s) condición(es) en la(s) que se requiere al algoritmo, dejar de evolucionar y presentar la mejor solución encontrada hasta el momento. En la literatura se presentan algunas alternativas aplicables a este parámetro de entrada<sup>36</sup>.

**Para Experimentación.** De acuerdo a la naturaleza experimental de la presente investigación aplicada de pregrado, se considera el escenario de máxima iteración (**IAG**) como criterio de parada para la corrida del AG-sTSP, donde la magnitud de este parámetro de entrada al problema, también será sugerido de acuerdo al resultado de los experimentos a realizar para cada de las instancias en particular.

**7.1.4.2 Proceso de Cruce.** Para problemas de permutación que utilizan una representación secuencial de símbolos, como lo es en este caso, una lista que describe una secuencia de visita de ciudades (tour) a realizar por el agente viajero, los operadores genéticos clásicos son obsoletos para su proceso, ya que se generarían individuos no válidos que comprenderían subciclos dentro del tour.

---

<sup>35</sup> ALBA, Enrique; LAGUNA, Manuel; MARTÍ, Rafael. Métodos evolutivos en problemas de optimización. [En Línea] Disponible en URL: <<http://www.uv.es/rmarti/paper/cpapers.html>> En Ingeniería UC 10 (3), pp. 80-89. [Citado el 8 de Diciembre de 2013].

<sup>36</sup> ARAUJO, Lourdes; CERVIGÓN, Carlos. Algoritmos Evolutivos: Un Enfoque Práctico. 1 ed. México: Editorial Alfaomega grupo editor S.A., Abril, 2009. p. 68. ISBN: 978-84-7897-911-0. [Citado el 8 de Diciembre de 2013].

**Para Experimentación.** Con el fin de preservar la variedad genética de la población y como valor agregado por parte de la presente investigación de pregrado, se propone la utilización de forma probabilística de dos operadores genéticos de cruce modificados:

- ❖ Cruce por Orden (OX)<sup>37</sup> → Cruce por Orden Multipunto (MOX: *Multipoint Order Crossing*) y,
- ❖ Cruce por Ciclo (CX)<sup>38</sup> → Cruce por Ciclo Ordenado (OCX: *Ordered Cycle Crossing*).

**7.1.4.3 Proceso de Mutación.** A través de alteraciones aleatorias en la información genética de los cromosomas, la mutación es un proceso que garantiza la diversificación del espacio de búsqueda que progresivamente es delimitado por el proceso natural de evolución de una población (reproducción/cruce).

**Para Experimentación.** De acuerdo a lo anterior, se contempla la utilización del heurístico de mutación basado en la Inversión Simple (Holland, 1975. SIM: Simple Investment Mutation), con una tasa de mutación asociada a los niveles porcentuales máximos y mínimos, expuestos en la revisión corta de la literatura como de uso común y sugerido para el buen desempeño de la metaheurística propuesta en la presente investigación aplicada de pregrado.

Remítase al Anexo D, para la consulta de las probabilidades de cruce y mutación de común sugerencia en la literatura revisada.

**7.1.4.4 Proceso de Inserción Élite.**<sup>39</sup> El elitismo consiste en asegurar la supervivencia de los individuos que presentan la mejor configuración de caracteres heredables en la población de tratamiento, es decir, aquellos cuyos valores de aptitud son los mejores según su evaluación en la función de adaptabilidad.

**Para Experimentación.** Se empleará de manera cuidadosa y probabilística, una lista élite que archive el registro del(os) mejor(es) ejemplar(es) de la población, de forma tal, que su inserción dado el caso de que en la nueva generación *no se llegara a presentar un cromosoma de mejor Fitness, y satisfecha la probabilidad de Inserción élite*, se efectúe el remplazo de aquel individuo que describa el peor Fitness en la población concurrente; lo anterior, con el fin de hacerlo(s) participe(s) de los procesos de selección, cruce y mutación para la producción de nuevos mejores individuos, esto, de acuerdo a la

---

<sup>37</sup> DAVIS, Lawrence. Applying Adaptive Algorithms to Epistatic Domains. 9th international joint conference on Artificial intelligence. August, 1985. Vol. 1. p. 162-164. ISBN: 0-934613-02-8 978-0-934-61302-6. [Citado el 8 de Diciembre de 2013].

<sup>38</sup> OLIVER, I.M.; SMITH, D.J.; HOLLAND, J.R.C. A Study of Permutation Crossover Operators on the Traveling Salesman Problem. Proceedings of the second International Conference on Genetic Algorithms, October, 1987. p. 224-233. ISBN: 0-8058-0158-8. [Citado el 8 de Diciembre de 2013].

<sup>39</sup> ARAUJO, Lourdes; CERVIGÓN, Carlos. Algoritmos Evolutivos: Un Enfoque Práctico. 1 ed. México: Editorial Alfaomega grupo editor S.A., Abril, 2009. p. 65. ISBN: 978-84-7897-911-0. [Citado el 8 de Diciembre de 2013].

conclusión presentada por De Jong<sup>40</sup> “El elitismo mejora la búsqueda local a expensas de la perspectiva global”.

## **7.2 DISEÑO DE UN ALGORITMO GENÉTICO PARA LA RESOLUCIÓN DE sTSP**

Para la fase de diseño del metaheurístico AG propuesto (AG-sTSP), planteado en la presente investigación aplicada de pregrado como una buena técnica para la búsqueda de una solución aproximada al sTSP; se tiene en cuenta la jerarquía y relación que presentan sus etapas identificadas en el análisis de la naturaleza de su pseudocódigo (Ver Inciso 7.1.1); para su codificación se hace uso del lenguaje de programación “Visual Basic para aplicaciones v.7.1” como extensión del programa Microsoft Excel.

A continuación, para ampliar la idea de su funcionamiento en el procesamiento de las instancias TSPLIB, se relacionan dichas etapas con la generalidad de su codificación:

### **7.2.1 Codificación de los Parámetros de Entrada del AG-sTSP.**

Remítase al Anexo E, para su consulta, y una mejor comprensión del inciso 7.2.

**7.2.2 Etapa 1: Generación de la Población de Tratamiento.** Para la generación de la población de tratamiento, como ha sido justificado anteriormente (Ver Inciso 7.1.2.3), se contempla la utilización de un individuo semilla de una calidad aparentemente buena, el cual, surge como producto de la implementación del Heurístico de Ahorros de Clarke & Wright.

**7.2.2.1 Individuo Semilla “Clarke & Wright”.** Su codificación comprende la siguiente dinámica:

1. Se usa la variable *Matriz\_Distancia()* para el cálculo y/o almacenamiento de los valores de las distancias euclidianas comprendidas entre cada uno de los lugares considerados en la instancia TSPLIB.
2. Se usa la variable *Matriz\_Ahorro()*, para el cálculo y almacenamiento de los ahorros de visitar una pareja de clientes diferentes con respecto al punto de partida, para esto, se utilizan las distancias suministradas por *Matriz\_Distancia()*; teniendo en cuenta, que el cálculo de ahorro de visita se hace mínimo (0 o un número negativo de acuerdo a la necesidad de la programación), en el caso de que la pareja de lugares considerada, esté compuesta por el mismo lugar, es decir:

$$\text{si } i = j ; \text{ entonces; } S_{ij} \leq 0.$$

3. Mientras se ejecuta el cálculo y almacenamiento de dichos ahorros, la introducción de dos diferentes bucles de búsquedas, facilitan encontrar en la matriz definida:

---

<sup>40</sup> DE JONG, Kenneth Alan. An Analysis of the Behavior of a Class of Genetic Adaptive Systems. PhD Thesis, Ann Arbor, MI, USA, 1975. [Citado el 8 de Diciembre de 2013].

- ❖ El mayor valor de ahorro  $Ahorro(0)$  y por ende el primer y principal tramo de tour factible  $De\_Tramo(0)$  y  $Hacia\_Tramo(0)$  Clarke and Wright (CnW).
  - ❖ El menor valor de ahorro  $VMin\_Ahorro$ , para el procedimiento de vetos de valores de la matriz de ahorros, y reinicio de los valores de ahorros extremos para la etapa de evaluación de los nuevos tramos factibles de ruta CnW.
4. Se emplean las variables  $XtrmIZQ$  y  $XtrmDER$  las cuales toman los valores extremos del tramo de ruta CnW en desarrollo, permitiendo la evaluación de los nuevos extremos de ruta por criterio de mayor ahorro generado; ya que a partir del principal tramo de tour factible CnW, sucesivamente se va añadiendo otro segmento de tour (en la parte izquierda o en la parte derecha de acuerdo a la relevancia que presente su ahorro en particular) hasta lograr su terminación.
  5. Se vetan valores de la matriz de ahorros mediante la utilización de la variable  $Veto\_Ahorro$ , al hacerla de una magnitud mucho menor, al valor mínimo de la matriz de ahorro encontrado anteriormente, es decir: " $Veto\_Ahorro = VMin\_Ahorro - 1$ ", esto, con el objeto de evitar la conformación de futuras subrutinas o anexo de tramos de ruta compuestas por 1 mismo lugar. Ya que la adición de los siguientes segmentos de tour factibles CnW, sucede de acuerdo a la elección de aquellos que proporcionan un mayor ahorro para construir la ruta CnW.
  6. Se insertan dos bucles de búsqueda para la etapa de evaluación de los nuevos tramos factibles de ruta CnW, los cuales encuentran el mayor valor de ahorro en la matriz de ahorros para su posterior comparación, teniendo en cuenta que:
    - ❖ El veto de valores de la matriz de ahorro, sucede antes y después de la búsqueda y comparación de los ahorros encontrados en la presente etapa.
    - ❖ Estando anclado el bucle de búsqueda a la variable  $XtrmIZQ$ , se encuentra su mayor valor de ahorro en  $Ahorro\_IZQ$  y posible tramo de ruta  $Nuevo\_De$ .
    - ❖ Estando anclado el bucle de búsqueda a la variable  $XtrmDER$ , se encuentra su mayor valor de ahorro en  $Ahorro\_DER$  y posible tramo de ruta  $Nuevo\_Hacia$ .

Una vez se obtienen los resultados en los bucles de búsqueda, estos se comparan para determinar cuál es el siguiente segmento factible de ruta CnW hasta su finalización, dichos valores se almacenan en las variables  $Ahorro()$ ,  $De\_Tramo()$  y  $Hacia\_Tramo()$ , estableciendo de esta forma, el individuo antepasado común o individuo semilla ( $Indiv\_CnW$ ) para la conformación de la población inicial de tratamiento para el AG.

**7.2.2.2 Alteración Probabilística del Individuo Semilla "BIT-SPLIT".** El objeto de este heurístico de transformación, es el de modificar  $n-1$  veces ( $n$  entendido como el valor que dimensiona el tamaño de la población de tratamiento para el AG) ( $n-1$  ya que se tiene en

cuenta la inclusión del *Indiv\_CnW* en la población inicial de tratamiento) el *Indiv\_CnW*, mediante el intercambio probabilístico de las posiciones que conforman su propia cadena de ADN o ruta factible. Su codificación presenta la siguiente dinámica:

1. Se fija una probabilidad de ocurrencia Bit-Split con una sensibilidad relativamente alta, la cual cumple con la función de condicionar la oportunidad de que suceda o no el intercambio de genes, dentro de la cadena de ADN del *Indiv\_CnW*.
2. Se contempla la utilización de una variable auxiliar  $K$ , la cual registra los valores 0, 1 y 2, cuando correspondientemente esta: es reiniciada, satisface la probabilidad Bit-Split por primera vez, y satisface la probabilidad Bit-Split por segunda vez.
3. Se hace uso de las variables de almacenamiento temporal, *Alelo1* para guardar el valor contenido en una posición específica del cromosoma (*Alelo*) y *Gen1* para guardar el valor de la posición del alelo (*Gen*) contenido en *Tacho1*.
4. Se evalúa la probabilidad Bit-Split en cada gen del cromosoma *Indiv\_CnW*, mediante la generación aleatoria de número decimal comprendido por los límites 0 y 1 (para su posterior mención, se asigna la anterior función a la variable *Aleatorio*), donde dado el caso de que:
  - ❖ No se satisfaga dicha condición, el gen y por lo tanto el alelo en consideración del *Indiv\_CnW*, es heredado automáticamente por el individuo en construcción para la constitución la población inicial de tratamiento.
  - ❖ Se satisfaga dicha condición con  $K=0$ ,  $K$  adopta el valor de 1 para tener entrada al procedimiento que le permite al proceso Bit-Split, determinar los valores de las variables *Alelo1* y *Gen1*.
  - ❖ Se satisfaga dicha condición con  $K=1$ ,  $K$  adopta el valor de 2 para tener acceso al procedimiento que le permite al proceso Bit-Split, ejecutar el remplazo del alelo actual en el gen almacenado anteriormente en *Gen1* y el remplazo del alelo almacenado anteriormente en *Alelo1* en el gen actual; finalmente,  $K$  adopta nuevamente el valor de 0, con el objeto de liberar el proceso Bit-Split para disponer del mismo mientras se recorre y evalúa la probabilidad Bit-Split a lo largo del cromosoma *Indiv\_CnW*.
5. Se aplica  $n-1$  veces al *Indiv\_CnW* el paso 4, para establecer la población inicial de tratamiento del AG, siendo esta almacenada temporalmente en la variable *Pob\_Trat1()*.

**7.2.3 Etapa 2: Evaluación de la Población de Tratamiento.** Desde la generalidad del asunto, esta etapa puede verse como el escenario que califica el grado de idoneidad que tiene la aplicación del proceso de evolución del AG-sTSP, en la transformación de la población de tratamiento; lo anterior, con el fin de determinar la viabilidad de llegar a implementar la mejor solución candidata encontrada hasta el momento en el problema en cuestión, para este caso el sTSP.

**7.2.3.1 Evaluación de la Población de Tratamiento.** Su codificación comprende la siguiente dinámica:

1. Como ha sido justificado anteriormente (Ver Inciso 7.1.3.2), para la evaluación de la población de tratamiento se contempla la inversa de la función objetivo para un problema de maximización, es decir:  $F(\mathbf{x}) = 1/f(\mathbf{x})$ ; lo cual resulta ideal para cuantificar la bondad de los cromosomas debido a la naturaleza de la optimización que en este caso, es la minimización del costo de recorrer (partiendo desde el depósito) todos los puntos de distribución (una vez cada uno), finalizando al regresar al punto de partida.
2. Se hace uso de la variable *Eval\_Pob()*, la cual, inicialmente, sirve para el almacenamiento de valores principales que se calculan en esta etapa, tales como: la aptitud, probabilidad de selección y puntaje acumulado de cada uno de los individuos de la población de tratamiento.
3. Se toma cada individuo almacenado en *Pob\_Trat1()*, para decodificar su *Fitness* mediante *Matriz\_Ahorro()* y almacenar los datos pertinentes en *Eval\_Pob()*.
4. Se calcula y se almacena temporalmente en la variable *Fitness\_Media*, el valor de la adaptabilidad media de la población de tratamiento; pues este un dato que sirve para lograr esbozar el comportamiento aptitudinal de la población de tratamiento a lo largo de la evolución del AG.

**7.2.3.2 Identificación de la Población Élite Inicial.** Aprovechando la “evaluación de la población inicial de tratamiento”, El objeto de este procedimiento es el de identificar y almacenar temporalmente los individuos élite de dicha población (ancestros élite), por tanto, una vez haya sido aplicado, este procedimiento caduca durante la corrida del AG. Su codificación comprende la siguiente dinámica:

1. Se hace uso de la variable *Indiv\_Elite1()* para almacenar en este caso, los ancestros élite (posición del ancestro élite en la población inicial, genotipo y *Fitness*).
3. Se utiliza un bucle de búsqueda que identifica dentro de la variable *Eval\_Pob()*, el mejor *Fitness* de la población inicial, y dentro del bucle de búsqueda, un bucle de comparación que identifica si el *Fitness* del Individuo ancestro considerado, ha sido almacenado con anterioridad en *Indiv\_Elite1()*, si el resultado es falso, Se almacena temporalmente en *Indiv\_Elite1()*: la posición que ocupa el Individuo ancestro considerado en *Pob\_Trat1()*, genotipo y *Fitness*; luego se aplica el paso 5; si el resultado es verdadero, Se ignora el Individuo ancestro considerado y se continua con el paso 4.
4. Se toma en consideración el siguiente *Fitness* de *Eval\_pob()* y se aplica el paso 3, dada la situación de que el 1er bucle de búsqueda no haya finalizado su ejecución, por otro lado si el 1er bucle ha finalizado la búsqueda, se aplica el paso 5.

5. Se finaliza este procedimiento si la variable *Indiv\_Elite1()* no tiene espacios disponibles para el almacenamiento temporal de información; pero si tiene, se considera el espacio disponible para almacenamiento y se aplica el paso 3.

**7.2.3.3 Selección Estocástica de Supervivientes de la Población de Tratamiento “Método de la Ruleta”.** Considerada como la versión estocástica de la supervivencia del más apto, en palabras coloquiales esta técnica puede ser descrita como la asignación de un segmento de ruleta a cada uno de los individuos de la población de tratamiento, en función de la relación existente entre su aptitud y la aptitud total de la población (probabilidad de selección). Su codificación comprende la siguiente dinámica:

1. Se extrae de la variable *Eval\_Pob()*, información que especifica la posición del segmento de ruleta que le fue asignado a cada uno de los individuos de la población de tratamiento, es decir, el puntaje acumulado de cada cromosoma.
2. Se hace uso de *Aleatorio*, para simular la elección de la posición del segmento de ruleta asignado a un individuo dado de la población de tratamiento.
3. Se hace uso de la variable *Pob\_Trat2()* que servirá para hacer el traspaso de los individuos seleccionados desde *Pob\_Trat1()*.
4. Se establece un bucle de búsqueda que tiene por objeto el identificar aquel segmento de ruleta elegido por *Aleatorio* (individuo de *Pob\_Trat1()*), dando origen a la supervivencia (cuando se elige un individuo X por 1era vez) o reproducción (Copia: cuando se elige un individuo X por más de 1 vez) de los individuos de la población de tratamiento.
5. Se almacena en *Pob\_trat2()* el individuo seleccionado.
6. Se aplica este procedimiento (paso 4) hasta lograr un número de selección de individuos igual al valor del parámetro *Tam\_Pob*.

**7.2.4 Etapa 3: Evolución de la Población de Tratamiento.** La presente etapa en el AG, tiene como objeto principal: transformar una proporción definida de la población de tratamiento, mediante la utilización de técnicas versátiles que combinan estrategias de intensificación en la exploración de la región de soluciones y diversificación en la explotación del espacio de soluciones, para finalmente encontrar soluciones de calidad relativamente buena al Problema del Agente Viajero Simétrico; Cabe anotar, que se contempla un enfoque generacional para el remplazo de los individuos de las poblaciones de tratamiento del AG-sTSP, en donde los individuos generados ya sea por cruce o mutación, se instalan en los lugares que ocupaban sus ascendentes inmediatos (cromosomas padres) en la población de tratamiento considerada.

**7.2.4.1 Cruce Probabilístico de Individuos de la Población de Tratamiento.** La idea de un cruce probabilístico para efectos de experimentación, es la de someter (cada vez que se quiera generar individuos descendientes) a su selección aleatoria, los dos

diferentes tipos de cruce propuestos (MOX y COX) , teniendo en cuenta que inicialmente los mismos, poseen igual probabilidad de ser elegidos pero con la peculiaridad del que el uso de los operadores de cruce queda restringido a generar máximo el 60% de individuos que deben ser generados a partir del proceso de cruce probabilístico; la particularidad de este proceso en la generación de nuevos individuos descendientes a partir de la población de tratamiento, puede ser descrita por la siguiente dinámica de su codificación:

1. Se elige de forma aleatoria de la población de tratamiento *Pob\_trat2()*, dos padres que no hayan sido utilizados con anterioridad por alguno de los procesos de cruce probabilísticos OCX o MOX, para esto: se compara la variable *Padres()*, que guarda temporalmente información sobre los padres elegidos de la población, con la variable *Veto\_Padres()*, que guarda información sobre los padres utilizados por alguno de los dos procesos de cruce precedentemente mencionados; entonces, si *Padres() = Veto\_Padres()*, dicho padre es reiniciado hasta ser diferente, cuando *Padres() ≠ Veto\_Padres()*, entonces *Padres()* se tienen en cuenta en la generación de los individuos descendientes a partir del operador de cruce que vaya a ser elegido, por tanto *Padres()* es almacenado temporalmente en *Veto\_Padres()*.

2. Se emplea *Aleatorio* para evaluar la probabilidad de apareamiento (*Prob\_Cruce*) entre los padres que satisfacen la condición expuesta anteriormente. Dado el caso de que no se dé el cruce entre los padres seleccionados, *Veto\_Padres()* olvida los padres almacenados recientemente, repitiéndose el paso anterior (1); en caso contrario, se procede al siguiente paso (3).

3. Se hace uso de las variables contadoras del tipo de cruce: *Contador\_Cruce\_OCX* y *Contador\_Cruce\_MOX*, con el fin de asegurar la participación máxima y mínima de los 2 tipos de cruce en la generación de los nuevos individuos descendientes por apareamiento (*Prop\_Pob\_Cruce*); lo anterior se logra mediante el cumplimiento de la siguiente condición: una vez la población de cruce generada por medio del cruce "X", sea mayor o igual al 60% de la que se debe producir por apareamiento, el 40% restante será producido por el cruce "Y". Si dicha situación aún no se presenta se procede con el paso 4.

4. *Aleatorio* evalúa la probabilidad de ocurrencia (selección) entre OCX y MOX para redireccionar el proceso de cruce probabilístico hacia el operador de cruce seleccionado en la aplicación para la generación de los individuos descendientes.

5. Se toma en cuenta el valor de la variable *Pob\_Cruce* que se actualiza en la interacción de los operadores de cruce, la cual identifica el número de individuos descendientes aportados mediante la interacción los padres de la población de tratamiento anterior y los operadores de cruce, en la constitución de la nueva población de tratamiento.

6. Se evalúa la situación:  $iPob\_Cruce \geq Prop\_Pob\_Cruce?$ , si resulta ser cierta, se finaliza el Proceso de **Cruce Probabilístico de Individuos de la Población de**

**Tratamiento, pero si resulta ser falsa, se continua con el paso 1 hasta que la evaluación en el paso 6 tenga como resultado ser verdadera.**

**7.2.4.1.1 Cruce Probabilístico por Cruce Cíclico Ordenado (OCX – Ordered Cycle Crossing).** En contraste al cruce MOX, este operador de cruce procesa de manera simultánea la información suministrada por los individuos padres para la constitución de los individuos descendientes, la dinámica de su codificación se presenta a continuación:

1. Se hace uso de la variable *Contador\_Cruce\_OCX*, la cual tiene por objeto contar el número de veces que este operador es aplicado a la población de tratamiento de una generación determinada; lo anterior para asegurar la implementación de la propuesta de uso restringido del operador de cruce (Ver Inciso 7.2.4.1, Paso 3).
2. Se hace uso de la variable *Reinicio\_Cruce\_OCX*, que funciona como una llave de salida del operador de cruce OCX, llevando cuenta de la no pertinencia de su utilización en la generación de individuos para la nueva población de tratamiento, para efectos de posterior conteo, dicha variable es inicializada como *Reinicio\_Cruce\_OCX = 0*.
3. Se hace uso de la variable *Num\_Ciclos*, que tiene por objeto identificar el número de ciclos de que se definen por las posiciones correspondientes de los valores, entre los individuos padres elegidos; dicha variable es inicializada como *Num\_Ciclos = 1*, ya que como mínimo, existe 1 ciclo definido entre los cromosomas padres elegidos.
4. Se hace uso de las variables: *Ini\_Ciclo* la cual guarda el alelo del gen que denota el inicio del ciclo considerado y *Fin\_Ciclo* guarda progresivamente el alelo del gen que potencialmente, denota el fin del ciclo considerado.
5. Se hace uso de la variable *Pos\_Ciclos()*, la cual guarda información sobre los genes que denotan el inicio de los ciclos revelados entre los cromosomas padres elegidos.
6. Se establece un bucle principal de comparación que va descubriendo los ciclos contenidos entre los dos cromosomas padres mediante la evaluación de la situación *¿Ini\_Ciclo ≠ Fin\_Ciclo?*; a medida que posicionalmente el operador de cruce recorre dichos cromosomas y el resultado de tal bucle es verdadero, los genes considerados van siendo asimilados respectivamente por la variable *Veto\_Ciclo()*, por tanto, *Veto\_Ciclo()* expone tanto la continuidad del ciclo en cuestión entre la pareja cromosomas como la necesidad de detección de los mismos; pero si el resultado de dicha comparación es falso, mediante un bucle de búsqueda se identifica de manera ordenada en *Veto\_Ciclo()* la existencia de algún gen disponible a partir de la evaluación de la situación *¿Veto\_Ciclo() = 0?*, si el resultado es verdadero, el gen disponible se almacena en *Pos\_Ciclo()*, se actualiza la variable *Num\_Ciclos = (Num\_Ciclos + 1)* y se actualiza *Ini\_Ciclo* al asignársele el alelo del gen disponible, dando vigencia a este “Procedimiento de Detección de Ciclos entre Individuos Padres”, es decir, se continua con la aplicación del paso 6; pero si el resultado es falso, este procedimiento se finaliza y se continua con el paso 7.

7. Se evalúa la situación  $\text{¿Num\_Ciclos} < 4?$  (donde 4 es el número mínimo de ciclos fijado para el funcionamiento regular del operador de cruce OCX, pues como se expone en el paso 9, 2 ciclos elegidos aleatoriamente son sucedidos automáticamente a los cromosomas descendientes), si el resultado es verdadero, se actualiza la variable  $\text{Reinicio\_Cruce\_OCX} = (\text{Reinicio\_Cruce\_OCX} + 1)$  y se continua con el paso 8; si el resultado es falso se continua con el paso 9.

8. Se evalúa la situación  $\text{¿Reinicio\_Cruce\_OCX} < 4?$  (donde 4 es la tolerancia fijada a conveniencia por motivos de ahorro en tiempo computacional, para soportar el reinicio del cruce OCX durante la corrida del AG-sTSP), si el resultado es verdadero, se reinicializa la elección aleatoria de los cromosomas padres junto con el respectivo *Veto\_Padre()* continuando con el paso 6; pero si el resultado es falso, se califica como no pertinente la utilización del cruce OCX, se actualiza la variable  $\text{Contador\_Cruce\_OCX} = (\text{Contador\_Cruce\_OCX} - 1)$  y se genera la salida del operador de cruce OCX.

9. Se eligen de forma aleatoria de acuerdo a la variable *Num\_Ciclos*, dos ciclos diferentes (*Ciclo1* y *Ciclo2*), para conformar el material genético que sucede automáticamente y sin cambio alguno, desde los individuos padres a los cromosomas descendientes.

10. Se identifica dentro de los cromosomas descendientes los genes disponibles para la sucesión ordenada y de forma inversa, del material genético restante (ciclos restantes) proveniente de los padres, para esto, se establece la posición del primer ciclo heredado a través de la evaluación de la situación  $\text{¿Pos\_Ciclo}(\text{Ciclo1}) = 1?$ , si resulta ser verdadero, la sucesión de los ciclos restantes, se hace en los genes disponibles ubicados después de  $\text{Pos\_Ciclo}(\text{Ciclo1})$ ; si resulta ser falso, también aplica la regla anterior, solo que en segundo lugar, también se tiene en cuenta los genes disponibles ubicados desde la primera posición hasta antes de  $\text{Pos\_Ciclo}(\text{Ciclo1})$ .

11. Se actualiza el valor de la variable  $\text{Pob\_Cruce} = (\text{Pob\_Cruce} + 2)$  y se evalúa la situación  $\text{¿Pob\_Cruce} < \text{Prop\_Pob\_Cruce}?$ , si resulta ser verdadero, se toman en cuenta los dos individuos descendientes generados por OCX; pero si resulta ser falso, se elige probabilísticamente uno de los dos cromosomas descendientes para su participación en la constitución de la nueva población de tratamiento.

12. Se finaliza la acción del operador de cruce OCX.

**7.2.4.1.2 Cruce Probabilístico por Cruce Ordenado Multipunto (MOX – Multipoint Ordered Crossing).** Este operador de cruce posee la particularidad de procesar de manera aislada, la información necesaria para la constitución de los individuos descendientes uno a la vez. La dinámica de su codificación se presenta a continuación:

1. Se hace uso de la variable *Contador\_Cruce\_MOX*, (comparte el mismo objeto de la variable *Contador\_Cruce\_OCX*).

2. Se hace uso de la variable *Distancia\_PCruce* que guarda el valor de la distancia mínima que debe existir entre los segmentos heredados de las cadenas de AND de los cromosomas padres.
3. Se hace uso de la variable *Prop\_Cadena\_ADN* para indicar el número máximo de genes que se contemplan para su compartición entre las adiciones de los puntos de cruce: *Adi\_P1* y *Adi\_P2*.
4. Se Hace uso de la variables *Punto\_Cruce1* y *Punto\_Cruce2* cuyos valores diferentes, generados de forma aleatoria (Rango de valores: Min = 1, Max = *P\_Distrib*), identifican los genes donde a partir de estos se añaden las adiciones correspondientes.
5. Se utilizan las variables *Adi\_P1* y *Adi\_P2* cuyos valores generados de forma aleatoria (Rango de valores *Adi\_P1*: Min = 1, Max = (*Prop\_Cadena\_ADN*-1); Rango de valores *Adi\_P2*: Min = 1, Max = (*Prop\_Cadena\_ADN*-*Adi\_P1*)) respectivamente junto con *Punto\_Cruce1* y *Punto\_Cruce2*, describen los arreglos del material genético que los cromosomas descendientes heredan automáticamente de los cromosomas padres.
6. Se reorganizan los puntos de cruces junto con sus adiciones correspondientes dado el caso de que se presente alguna de las siguientes situaciones:
  - a.  $Punto\_Cruce1 > Punto\_Cruce2$ .
  - b.  $(Punto\_Cruce1 + Adi\_P1) = P\_Distrib$ .
  - c.  $(Punto\_Cruce1 + Adi\_P1) > P\_Distrib$ .
  - d.  $Punto\_Cruce2 > P\_Distrib$ .
  - e.  $Punto\_Cruce2 = P\_Distrib$  y  $(Adi\_P2 + Distancia\_PCruce) \geq Punto\_Cruce1$
  - f.  $Punto\_Cruce2 < P\_Distrib$  pero  $(Punto\_Cruce2 + Adi\_P2) > P\_Distrib$
7. Se distingue el individuo descendiente a generar mediante el operador de cruce, a través de la variable *DESCENDIENTE*, inicialmente *DESCENDIENTE* = 1.
8. Se realiza la sucesión del material genético heredable desde el respectivo padre principal (puntos de cruce y adiciones), al individuo *DESCENDIENTE*.
9. Se emplea la variable auxiliar *N*, la cual, establece un punto de impresión para los genes restantes del padre secundario de acuerdo al resultado de la evaluación de la situación: ¿ $(Punto\_Cruce2 + Adi\_P2) \geq P\_Distrib$ ?, si resulta ser cierta,  $N = (Punto\_Cruce1 + Adi\_P1 + 1)$ ; **pero si resulta falsa,  $N = (Punto\_Cruce2 + Adi\_P2 + 1)$ .**
10. Se imprimen en el individuo descendiente, a partir del gen *N*, aquellos alelos (valores) del padre secundario que comparados con los existentes en el individuo descendiente, resultan ser diferentes; dicha comparación e impresión de alelos se hace de forma ordenada, es decir:

**10.1** De  $N$  a  $P\_Distrib$  y luego de  $1$  a  $N$  pero con la peculiaridad de que para efecto de reutilización de variable,  $N$  toma otro valor de acuerdo al resultado de la evaluación de la situación  $\{ (Punto\_Cruce2 + Adi\_P2) \geq P\_Distrib \}$ ; si resulta ser cierta,  $N = (Punto\_Cruce2 - 1)$ ; pero si resulta ser falsa,  $N = (Punto\_Cruce1 - 1)$ .

**11.** Se actualiza el valor de la variable  $Pob\_Cruce = (Pob\_Cruce + 1)$ , luego se identifica  $\{ DESCENDIENTE = 2 \}$ , si resulta ser verdadero, se finaliza la acción del operador de cruce MOX; pero si resulta ser falso, se continua con el paso 12.

**12.** Se evalúa la situación:  $\{ Pob\_Cruce < Prop\_Pob\_Cruce \}$ , si resulta verdadero, se actualiza la variable  $DESCENDIENTE = 2$  y se continua con el paso 8 para la generación del individuo identificado por  $DESCENDIENTE$ ; pero si resulta falso, se finaliza la acción del operador de cruce MOX.

**7.2.4.2 Mutación SIM de la Nueva Población de Tratamiento Restante.** Como parte de esta propuesta de investigación se contempla para el AG diseñado, el aplicar el proceso de "Mutación basada en la Inversión Simple" (SIM) sobre la nueva población de tratamiento exceptuando aquellos individuos que fueron empleados en los procesos de cruces probabilísticos para la creación de nuevos individuos descendientes; su objeto es el de generar cambios aleatorios (casi imperceptibles) en el material genético de los individuos considerados, imitando según "la teoría de la evolución biológica de Darwin", las pequeñas variaciones morfológicas que se presentan en los individuos de una misma especie. La dinámica de su codificación es la siguiente:

**1.** Se hace uso de la variable  $Prop\_Cadena\_Mutacion$ , la cual indica el número de genes de un cromosoma de la población que serán tratados por el proceso de alteración que tiene como valor mínimo 2.

**2.** Se hace uso de la variable  $Punto\_Mutacion$ , la cual señala el gen donde a partir del mismo se da el proceso de alteración SIM.

**3.** Se hace uso de la variable  $Pob\_Mutacion$ , la cual identifica el número de individuos transformados de la nueva población de tratamiento disponible, mediante el proceso de mutación SIM.

**4.** Se hace uso de la variable  $Indiv\_Disponibles()$ , la cual guarda información que identifica aquellos individuos disponibles de la nueva población de tratamiento, para su posible mutación. Esto, para generar un ahorro de tiempo computacional para el AG ya que la elección del individuo (el cual debe ser diferente a los previamente empleados en los procesos de cruce probabilísticos) y su punto de mutación no garantizan la aplicación del procedimiento SIM.

**5.** Se establece un bucle de comparación que inicialmente nombra un individuo a partir de la variable  $Selecc\_Indiv$  para su comparación con cada uno de los individuos almacenados en  $Veto\_Padres()$ , por tanto, se evalúa la situación

$\text{¿Selecc\_Indiv} = \text{Veto\_Padres}()$ ?, si en algún momento de la comparación resulta cierto, se actualiza  $\text{Selecc\_Indiv} = (\text{Selecc\_Indiv} + 1)$  y se continua con el paso 5; si al terminar la comparación resulta falso,  $\text{Indiv\_Disponibles}() = \text{Selecc\_Indiv}$  y se continua con el paso 6.

6. Se averigua si aún queda espacio de almacenamiento en  $\text{Indiv\_Disponibles}()$ , si hay espacio de almacenamiento en la variable, se aplica el paso 5; pero si no hay espacio de almacenamiento en la variable, se continua con el paso 7.

7. Se reutiliza la variable  $\text{Selecc\_Indiv}$  para elegir de manera aleatoria de la nueva población de tratamiento disponible almacenada en  $\text{Indiv\_Disponibles}()$ , el individuo a transformar mediante SIM, luego mediante un bucle de comparación se comprueba en  $\text{Veto\_Padres}()$  (a partir del espacio  $\text{Prop\_Pob\_Cruce} + 1$ ) en los espacios restantes la siguiente situación,  $\text{¿Veto\_Padres}() = \text{Selecc\_Indiv} ?$ , si en algún momento de la comparación resulta ser cierto, se reinicia  $\text{Selecc\_Indiv}$  repitiéndose el paso 7; si al terminar la comparación resulta ser falso, se considera el individuo seleccionado y se continua con el paso 8.

8. Se elige mediante  $\text{Aleatorio}$ , el punto de mutación dentro del cromosoma seleccionado para efectuar la alteración SIM y se evalúa la probabilidad de que el cromosoma elegido mute mediante la evaluación de la situación  $\text{¿Aleatorio} \leq \text{Prob\_Mutacion}?$ , si resulta falso, se continua con el paso 11; si resulta cierto, se continua con el paso 9.

9. Se establece un bucle de búsqueda que va copiando de  $\text{Pob\_Trat2}()$ , el individuo elegido hasta el momento en el que se identifica su punto de mutación (gen a partir del cual se efectúa la alteración SIM); una vez encontrado el dicho punto de mutación, se invierte el segmento de cromosoma comprendido entre  $\text{Punto\_Mutacion}$  y  $(\text{Punto\_Mutacion} + \text{Prop\_Cadena\_Mutacion})$ ; tal segmento invertido se copia en el individuo en proceso formación y luego, si queda material genético por copiar, se hace la transferencia de información y se pasa al paso 10; si no queda material genético restante por copiar, simplemente se continua con el paso 10.

10. Se actualiza la variable  $\text{Pob\_Mutacion} = (\text{Pob\_Mutacion} + 1)$  y se veta el individuo mutado para evitar su posterior reutilización mediante su inclusión en la variable  $\text{Veto\_Padres}() = \text{Selecc\_Indiv}$ , (a partir de  $\text{Prop\_Pob\_Cruce} + 1$  hasta  $\text{Prop\_Pob\_Cruce} + \text{Prop\_Pob\_Mutacion}$ ).

11. Se evalúa la situación  $\text{¿Pob\_Mutación} = \text{Prop\_Pob\_Mutada}?$ , si resulta ser falso, se continua con el paso 7; si resulta ser verdadero, se finaliza la Mutación SIM de la Nueva Población de Tratamiento Restante.

**7.2.4.3 Inserción de Individuos Élite sobre la Nueva Población de Tratamiento.** Este procedimiento se aplica a la nueva población de tratamiento en general; Con el fin de darles la oportunidad de poder participar en posteriores etapas de evolución de la población de tratamiento y teniendo en cuenta el efecto adverso de su utilización abierta

en el AG (convergencia prematura del AG), su inclusión en la población de tratamiento puede verse mitigada por varias razones, algunas de estas serían: por la probabilidad de su ocurrencia la cual es evaluada en su propio procedimiento (Ver Inciso 7.2.4.3.2), por la probabilidad de su supervivencia la cual es puesta a prueba en el proceso de selección natural (método de la ruleta), etc. A continuación se presenta la dinámica de su codificación:

#### **7.2.4.3.1 Identificación de la Nueva Población de Élite e Inadaptados.**

1. Se hace uso de la variable *Indiv\_Elite2()* para el almacenamiento temporal de la siguiente información: posición del(los) individuo(s) élite(s) en la nueva población de tratamiento *Pob\_Trat2()*, genotipo(s) y respectivo(s) *Fitness* (valor de aptitud o adaptabilidad).
2. Se hace uso de la variable *Inadaptados()* para el almacenamiento temporal de la siguiente información: posición del(los) individuo(s) menos adaptado(s) en *Pob\_Trat2()* y respectivo(s) *Fitness* (valor de aptitud).
3. Se hace uso del procedimiento de evaluación de la población de tratamiento (Ver Inciso 7.2.3.1) donde se tiene en cuenta procesar en *Eval\_pob()*, únicamente y de manera ordenada, la adaptabilidad de cada uno de los individuos de la nueva población de tratamiento ubicados en la variable *Pob\_Trat2()*.
4. Se establece un bucle de comparación que evalúa sobre los nuevos valores de la variable *Eval\_Pob()*, la existencia de *Fitness* de mejor calidad; dentro de dicho bucle de comparación, se establece otro bucle de cotejo de valores de aptitud, el cual autoriza en *Indiv\_Elite2()*, la grabación de la información pertinente en relación al élite en cuestión (revisar paso 1), dado el caso de que el *Fitness* en consideración no se encuentre almacenado anteriormente en *Indiv\_Elite2()*. Se puede decir que este procedimiento posee una dinámica similar a la descrita en el procedimiento de "Identificación de la Población Élite Inicial" (Ver Inciso 7.2.3.2).
5. Se utilizan los mismos dos bucles mencionados en el paso 4 para identificar la población inadaptada (individuos de pésimo *Fitness*) dentro de la nueva población de tratamiento; ahora, con la particularidad de que el criterio de elección de los individuos que conforman dicha población está basado en el *Fitness* de peor calidad; y el almacenamiento de la información requerida, se hace en la variable *Inadaptados()*.
6. Para efectos de iteración del AG propuesto, se hace el siguiente traspaso de información entre las variables  $Pob\_Trat1() = Pob\_Trat2()$ , ya que en la Etapa 2 (Evaluación de la población de Tratamiento), está condicionado el uso principal de tal variable.

#### 7.2.4.3.2 Comparación de las Poblaciones Élite.

1. Se hace uso de la variable booleana *Tenencia\_Elite* para autorizar la liberación o retención del mejor individuo elite de la población anterior ( $Indiv\_Elite1(0)$ ).
2. Se hace uso de la variable *Retención* para indicar el estado del conteo de retención del mejor individuo elite de la población anterior.
3. Se hace uso de la variable auxiliar *M* para identificar el individuo de peor *Fitness* de la nueva población de tratamiento, el cual es tomado en cuenta para su posible remplazo de acuerdo al resultado del bucle de comparación de las poblaciones de individuos elites con un valor inicial de  $M = 0$ .
4. Se comprueba el estado de la variable booleana *Tenencia\_Elite*, si resulta ser verdadero (*True*), no se modifica la población elite anterior; pero si resulta ser falso (*False*), se actualiza el estado de la variable de retención del mejor individuo elite de la población anterior como  $Retención = (Retención + 1)$ .
5. Se evalúa la situación  $¿Retención = Max\_Retención?$ , si resulta ser falsa, se ignora el procedimiento y se continua con el paso 6; si resulta ser verdadero, se emplea un bucle de remplazo que elimina el mejor individuo elite de la población anterior dejando vacío aquel espacio de la variable que lo contiene ( $Indiv\_Elite1(0) = 0$ ), se reinicializa la variable  $Retención = 0$  y se actualiza la variable  $Tenencia\_Elite = True$ ; como consideración especial, si  $Pob\_Elite > 1$ , se promueve o se asciende la categoría de los demás individuos elites de las población de tratamiento anterior quedando vacío el último espacio de la variable que los contiene ( $Indiv\_Elite1(Pob\_Elite - 1) = 0$ ).
6. Se utiliza un bucle de comparación de las poblaciones elites anterior y nueva, el cual evalúa la situación  $¿Indiv\_Elite1(N) > Indiv\_Elite2(N)?$  donde el criterio tenido en cuenta es el *Fitness* de cada individuo; si resulta ser verdadero, se continua con el paso 7; si resulta ser falso se continua con el paso 8.
7. Se degrada de categoría al individuo elite de la nueva población (se habla en plural si  $Pob\_Elite > 1$ , por tanto en este caso, se descarta el ultimo individuo elite de dicha población), se emplea *Aleatorio* para evaluar la probabilidad de ocurrencia de la inserción del individuo elite de la población anterior en la nueva población de tratamiento  $¿Aleatorio <= Prob\_Elite?$ , si el resultado es falso, no ocurre inserción elite y se continua con el paso 9; si el resultado es verdadero se procede con el paso 7.1.

7.1 Se evalúa  $¿N = 0?$ , si el resultado es falso se procede con el paso 7.2; si el resultado es verdadero, se actualiza la variable  $Tenencia\_Elite = False$  y se procede con el paso 7.2

7.2 Se ejecuta dicha inserción mediante el remplazo entre el individuo elite y el individuo de peor *Fitness* de la nueva población de tratamiento de la forma:

$Pob\_Trat1(Inadaptados(M,0)) = Indiv\_Elite1()$ , se actualiza la variable  $M = (M + 1)$ , y se continua con el paso 9.

**8.** Se evalúa  $\zeta N = 0$ ? Si el resultado es verdadero Se actualiza la variable  $Tenencia\_Elite = True$ , la variable  $Retención = 0$  y se continua con 8.1; si es falso, se continua con el paso 8.1.

**8.1** Se degrada de categoría al individuo élite de la anterior población (se habla en plural si  $Pob\_Elite > 1$ , por tanto en este caso, se descarta el último individuo élite de dicha población), para hacer espacio al nuevo mejor individuo élite de la población, en este caso la inserción élite se da de la forma  $Indiv\_Elite1() = Indiv\_Elite2()$ , se procede con el paso 9.

**9.** Se actualiza  $N = (N + 1)$ , y se comprueba si el bucle de comparación definido en el paso 5 ha terminado de cotejar todos los individuos élite de la anterior población Vs. Los individuos élite que se puedan de la nueva población; si no ha terminado el cotejo, se continúa con el paso 5; si ha terminado el cotejo, Se finaliza el procedimiento de Inserción de Individuos Élite sobre la Nueva Población de Tratamiento Restante.

**10.** Se actualiza el criterio de parada (Iteración o Tiempo) para evaluar la finalización de la corrida del Algoritmo Genético propuesto; si el criterio de parada no ha sido satisfecho, se contempla una nueva iteración del AG **hasta ser declarado el criterio de parada del AG como satisfecho; si el criterio de parada ha sido satisfecho, se finaliza el AG-sTSP para presentar los resultados del procesamiento de la instancia sTSP considerada.**

### 7.3 IMPLEMENTACIÓN DEL AG DISEÑADO EN LA RESOLUCIÓN DEL sTSP

**7.3.1 Experimentación.** Con el fin de presentar soluciones aproximadas y replicables por medio del AG-sTSP a las instancias sTSP TSPLIB, se adopta la estrategia de experimentar de acuerdo al lineamiento que presenta un arreglo factorial completo  $2^k$  replicado, considerándose como una estrategia experimental óptima para el estudio simultáneo del efecto de los factores y sus interacciones<sup>41</sup> -entendiéndose como interacción: “cuando la forma en la que un factor influye en la variable de respuesta, está afectada por el nivel en el que se encuentre otro factor”-.

Aunque la naturaleza de este tipo de experimentación resulta ser un proceso costoso de realizar debido a que se contemplan todas las posibles combinaciones de los factores que intervienen en el fenómeno a estudiar, además de su replicación; se adopta esta estrategia por razón de que los recursos requeridos para experimentación (Laboratorio de Simulación UPB Bucaramanga), se encuentran a disposición del experimentador.

---

<sup>41</sup> MEDINA V. Pedro Daniel; LOPEZ R. Ángela María. Análisis Crítico del Diseño Factorial  $2^k$  Sobre Casos Aplicados. Scientia et Technica Año XVIII, No 47, Abril de 2011. Universidad Tecnológica de Pereira. p. 101-106. ISSN 0122-1701 [Citado el 20 de Junio de 2013].

**7.3.1.1 Características del Ordenador.** Con el fin de mantener una mayor uniformidad en el tiempo de procesamiento de las instancias por parte del AG propuesto (AG-sTSP), para el desarrollo de los experimentos elementales se hizo uso de un conjunto de ordenadores que contaban con la siguiente descripción básica de sistema:

<b>Edición Windows:</b>	Windows 7 Enterprise.
<b>Procesador:</b>	Intel(R) Core(TM) i7-2600 CPU @ 3.40GHz.
<b>Memoria RAM instalada:</b>	8,00 GB (3,24 GB Utilizable).
<b>Tipo de Sistema:</b>	Sistema Operativo de 32 bits.

**7.3.1.2 Instancias sTSP-TSPLIB.** Prof. Dr. Gerhard Reinelt del “Institut für Angewandte Mathematik” de UNIVERSITÄT HEIDELBERG en Alemania, acopia y conserva una librería de dominio público llamada TSPLIB<sup>42</sup>, en la cual se pueden encontrar instancias de ejemplo para el TSP (y problemas relacionados) de varias fuentes y varios tipos.

A continuación en la Tabla 4, Se referencian las Instancias de la clase sTSP que fueron empleadas en el desarrollo de los experimentos elementales de la presente investigación aplicada de pregrado para efectos de posterior comparación con los resultados obtenidos por la investigación “Un Enfoque de Búsqueda Tabú para el Problema del Agente Viajero” del autor Jhon Gabriel Guerrero Becerra de la Universidad Pontificia Bolivariana seccional Bucaramanga – Colombia (Ts-TSP UPB) y la solución óptima hasta el momento suministrada por TSPLIB (Z OPT.):

**Tabla 4. Instancias sTSP de TSPLIB Procesadas en el AG-sTSP.**

NOMBRE DE LA INSTANCIA	Z OPT.
<i>bays29</i>	2020
<i>eil51</i>	426
<i>st70</i>	675
<i>eil76</i>	538
<i>pr76</i>	108159

**Fuente: Elaborado por el Autor.**

**7.3.1.3 Identificación de Factores.** A continuación en la tabla 5. Se identifica el tipo de variable, Nombre de los factores y su respectiva codificación asignada por el autor en el AG-sTSP para efectos de experimentación.

<sup>42</sup> Discrete and Combinatorial Optimization. TSPLIB, Public Domain Library. [On Line] URL: <<http://www1.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/index.html>>

**Tabla 5. Variables para el Arreglo Experimental Factorial 2<sup>k</sup> Completo.**

TIPO DE VARIABLE	NOMBRE DEL FACTOR	ABREV.	CODIFICACIÓN AG-sTSP
Independiente	Adaptabilidad del Individuo	<i>Fitness</i>	<i>Fitness</i>
Dependientes (Factores)	Tamaño de la Población de Tratamiento	<i>Tpob</i>	<i>Tam_Pob</i>
	Probabilidad de Cruce de Individuos	<i>Pc</i>	<i>Prob_Cruce</i>
	Probabilidad de Mutación del Individuo	<i>Pm</i>	<i>Prob_Mutacion</i>
	Número de Iteraciones para el AG	<i>IAG</i>	<i>Criterio_Parada_AG</i>
	Tenencia Elite	<i>Te</i>	<i>Tenencia_Elite</i>

**Fuente: Elaborado por el Autor.**

**7.3.1.4 Número de Experimentos.** Como se propone la estrategia de desarrollar los experimentos teniendo en cuenta un arreglo factorial completo 2<sup>k</sup> replicado, de 5 factores, replicado 4 veces; el # total de experimentos elementales para cada instancia TSPLIB, es de 128 y el # total de experimentos elementales por sesión para cada instancia TSPLIB, es de 64.

$$\text{Núm. Exp. Total} = (2^k) * R \quad (2)$$

$$\text{Núm. Exp. Sesión} = \left( \frac{\text{Núm. Exp. Total}}{2} \right) \quad (3)$$

Dónde:

*k* = Número de Factores.

*R* = Número de Réplicas del Experimento.

**Núm. Exp. Total** = Núm. total de experimentos elementales para cada instancia TSPLIB.

**Núm. Exp. Sesión** = Núm. total de experimentos elementales por sesión para cada instancia TSPLIB.

Cada sesión comprende una configuración tentativa diferente de los parámetros de entrada para el procesamiento de las instancias TSPLIB por medio del AG propuesto.

**7.3.2 Resultados Experimentales.** Para de esta sección en adelante, la expresión “mejores soluciones factibles” hace referencia a aquellas soluciones encontradas por el AG propuesto (AG-sTSP) que presentan un costo de ruta ≤ a los encontrados en “Ts-TSP UPB” para cada instancia en particular. Para efectos de posterior análisis, se toman en cuenta, de las mejores soluciones factibles de las 4 últimas sesiones experimentales de cada Instancia procesada por el AG-sTSP, lo siguiente:

- ❖ Por cada sesión de cada instancia procesada, un rango de resultados factibles el cual comprende: 3 soluciones factibles de la más baja calidad y 3 soluciones factibles de la más alta calidad,
- ❖ Las respectivas proporciones de las soluciones factibles de dicho rango, respecto a la población de soluciones de cada sesión “Prop AG-sTSP”,
- ❖ La comparación con el “Ts-TSP Gap” (Gap Búsqueda Tabú del autor Jhon Gabriel G. Becerra de la Universidad Pontificia Bolivariana – Bucaramanga - Colombia),
- ❖ La comparación con el “Z Opt Gap” (Gap del costo óptimo suministrado por TSPLIB) y
- ❖ La comparación con el “CnW Gap” (Gap del costo factible que describe la heurística de ahorros de Clarke and Wright).

A continuación, el “Gap” puede entenderse como una medida de razón porcentual que compara la calidad del costo de una solución encontrada, respecto a otra solución factible en consideración. Su interpretación depende del sentido y magnitud que adquiere su valor: Si es positivo (+), se evidencia la superioridad de la calidad de la solución encontrada frente a la calidad de la solución factible en consideración, caso contrario donde se evidencia la inferioridad de la calidad de la solución encontrada, el Gap es (-) negativo.

$$RTA C. Gap = \left( 100 * \left( \frac{RTA C. - RTA E.}{RTA C.} \right) \right) \% \quad (4)$$

Dónde:

*RTA E.* = Respuesta Factible Encontrada, *RTA C.* = Respuesta Factible Considerada.

**NOTA:**

Tener en cuenta que la métrica porcentual del Gap, toma como base para su cálculo la magnitud de la respuesta considerada (*RTA C.*).

Remítase a la carpeta de Anexo “00 Tabulación de Resultados AG para Instancias sTSP-TSPLIB”, desde el archivo 00 al 04 de extensión “.xlsx”, para una consulta completa de los resultados de los experimentos elementales de cada instancia TSPLIB según la estrategia del arreglo factorial completo  $2^k$  replicado 4 veces.

Remítase al Anexo F, para la consulta de las configuraciones tentativas de los parámetros de entrada del AG-sTSP y la consulta completa de las mejores soluciones factibles de las 4 últimas sesiones experimentales para cada instancia sTSP TSPLIB.

### 7.3.2.1 Instancia bays29.

Tabla 6. Referencias de Costos de Rutas Factibles de bays29.

COSTO DE RUTA FACTIBLE	
Z Opt	2020
Ts-TSP UPB	2026
CnW	2684

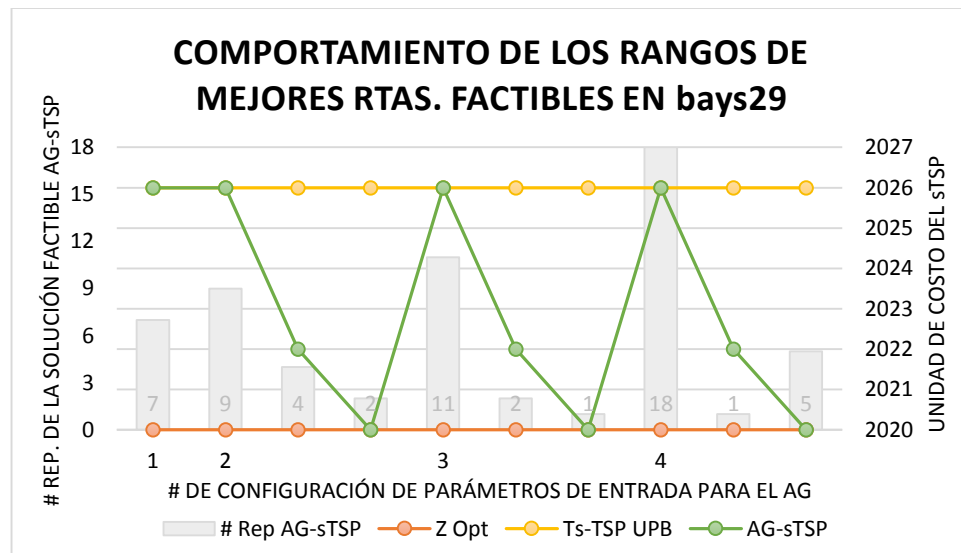
Fuente: Elaborado por el Autor.

Tabla 7. Rangos de Resultados Factibles de cada Sesión en la Instancia bays29.

PARÁMETROS	AG-sTSP	# Rep.	Prop AG-sTSP	Z Opt Gap	Ts-TSP Gap	CnW Gap
Configuración 1	2026	7	10,94%	-0,30%	0,00%	24,52%
Configuración 2	2026	9	14,06%	-0,30%	0,00%	24,52%
	2022	4	6,25%	-0,10%	0,20%	24,66%
Configuración 3	2020	2	3,13%	0,00%	0,30%	24,74%
	2026	11	17,19%	-0,30%	0,00%	24,52%
	2022	2	3,13%	-0,10%	0,20%	24,66%
Configuración 4	2020	1	1,56%	0,00%	0,30%	24,74%
	2026	18	28,13%	-0,30%	0,00%	24,52%
	2022	1	1,56%	-0,10%	0,20%	24,66%
	2020	5	7,81%	0,00%	0,30%	24,74%

Fuente: Elaborado por el Autor.

Gráfica 1. Comportamiento de los Rangos de Resultados Factibles en bays29.



Fuente: Elaborado por el Autor.

### 7.3.2.2 Instancia eil51.

Tabla 8. Referencias de Costos de Rutas Factibles de eil51.

COSTO DE RUTA FACTIBLE	
Z Opt	426
Ts-TSP UPB	455
CnW	544

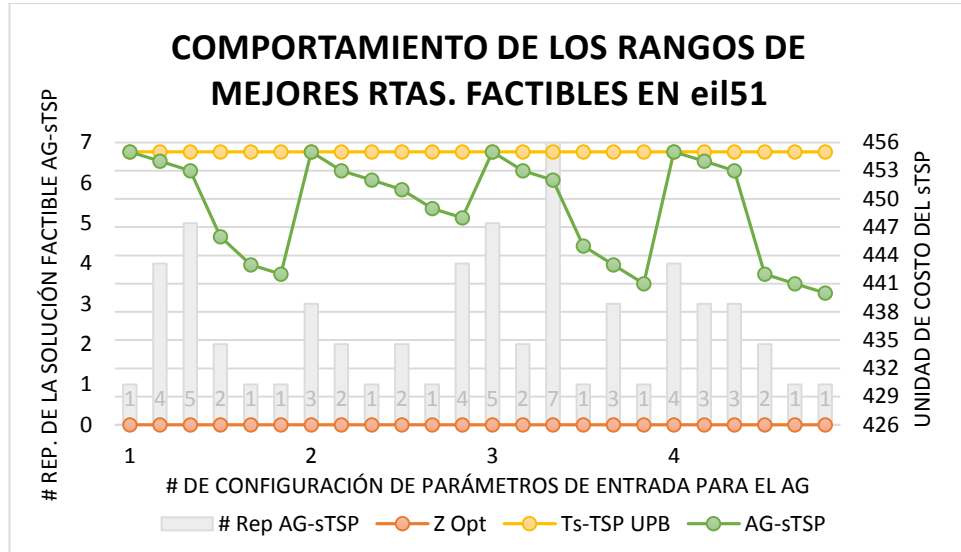
Fuente: Elaborado por el Autor.

Tabla 9. Rangos de Resultados Factibles de cada Sesión en la Instancia eil51.

PARÁMETROS	AG-sTSP	# Rep.	Prop AG-sTSP	Z Opt Gap	Ts-TSP Gap	CnW Gap
Configuración 1	455	1	1,56%	-6,81%	0,00%	16,36%
	454	4	6,25%	-6,57%	0,22%	16,54%
	453	5	7,81%	-6,34%	0,44%	16,73%
	446	2	3,13%	-4,69%	1,98%	18,01%
	443	1	1,56%	-3,99%	2,64%	18,57%
	442	1	1,56%	-3,76%	2,86%	18,75%
Configuración 2	455	3	4,69%	-6,81%	0,00%	16,36%
	453	2	3,13%	-6,34%	0,44%	16,73%
	452	1	1,56%	-6,10%	0,66%	16,91%
	451	2	3,13%	-5,87%	0,88%	17,10%
	449	1	1,56%	-5,40%	1,32%	17,46%
	448	4	6,25%	-5,16%	1,54%	17,65%
Configuración 3	455	5	7,81%	-6,81%	0,00%	16,36%
	453	2	3,13%	-6,34%	0,44%	16,73%
	452	7	10,94%	-6,10%	0,66%	16,91%
	445	1	1,56%	-4,46%	2,20%	18,20%
	443	3	4,69%	-3,99%	2,64%	18,57%
	441	1	1,56%	-3,52%	3,08%	18,93%
Configuración 4	455	4	6,25%	-6,81%	0,00%	16,36%
	454	3	4,69%	-6,57%	0,22%	16,54%
	453	3	4,69%	-6,34%	0,44%	16,73%
	442	2	3,13%	-3,76%	2,86%	18,75%
	441	1	1,56%	-3,52%	3,08%	18,93%
	440	1	1,56%	-3,29%	3,30%	19,12%

Fuente: Elaborado por el Autor.

Gráfica 2. Comportamiento de los Rangos de Resultados Factibles en ei51.



Fuente: Elaborado por el Autor.

### 7.3.2.3 Instancia st70.

Tabla 10. Referencias de Costos de Rutas Factibles de st70.

	COSTO DE RUTA FACTIBLE
Z Opt	675
Ts-TSP UPB	738
CnW	949

Fuente: Elaborado por el Autor.

Tabla 11. Rangos de Resultados Factibles de cada Sesión en la Instancia st70.

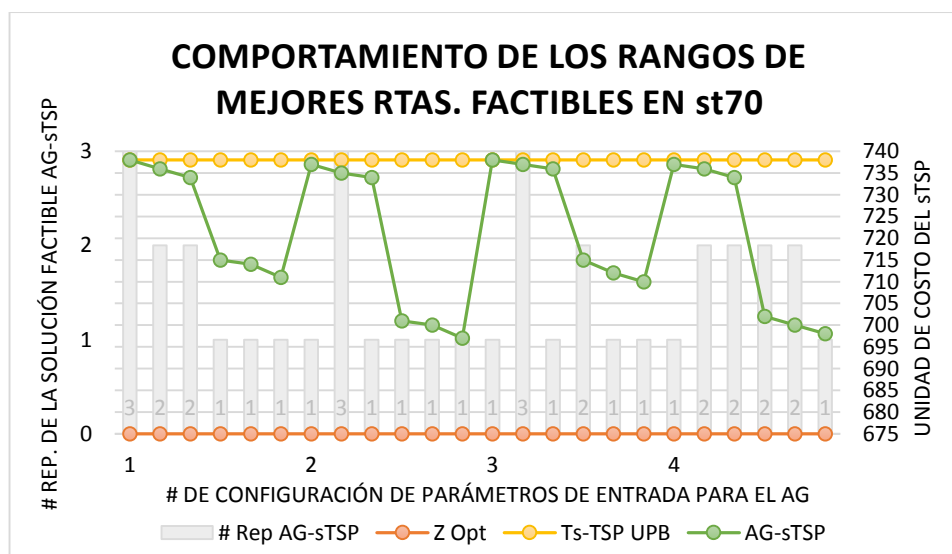
PARÁMETROS	AG-sTSP	# Rep.	Prop AG-sTSP	Z Opt Gap	Ts-TSP Gap	CnW Gap
Configuración 1	738	3	4,69%	-9,33%	0,00%	22,23%
	736	2	3,13%	-9,04%	0,27%	22,44%
	734	2	3,13%	-8,74%	0,54%	22,66%
	715	1	1,56%	-5,93%	3,12%	24,66%
	714	1	1,56%	-5,78%	3,25%	24,76%
	711	1	1,56%	-5,33%	3,66%	25,08%
Configuración 2	737	1	1,56%	-9,19%	0,14%	22,34%
	735	3	4,69%	-8,89%	0,41%	22,55%
	734	1	1,56%	-8,74%	0,54%	22,66%
	701	1	1,56%	-3,85%	5,01%	26,13%
	700	1	1,56%	-3,70%	5,15%	26,24%

**Tabla 11. Rangos de Resultados Factibles de cada Sesión en la Instancia st70. (Continuación)**

<b>Configuración 2</b>	697	1	1,56%	-3,26%	5,56%	26,55%
<b>Configuración 3</b>	738	1	1,56%	-9,33%	0,00%	22,23%
	737	3	4,69%	-9,19%	0,14%	22,34%
	736	1	1,56%	-9,04%	0,27%	22,44%
	715	2	3,13%	-5,93%	3,12%	24,66%
	712	1	1,56%	-5,48%	3,52%	24,97%
	710	1	1,56%	-5,19%	3,79%	25,18%
<b>Configuración 4</b>	737	1	1,56%	-9,19%	0,14%	22,34%
	736	2	3,13%	-9,04%	0,27%	22,44%
	734	2	3,13%	-8,74%	0,54%	22,66%
	702	2	3,13%	-4,00%	4,88%	26,03%
	700	2	3,13%	-3,70%	5,15%	26,24%
	698	1	1,56%	-3,41%	5,42%	26,45%

Fuente: Elaborado por el Autor.

**Gráfica 3. Comportamiento de los Rangos de Resultados Factibles en st70.**



Fuente: Elaborado por el Autor.

### 7.3.2.4 Instancia eil76.

Tabla 12. Referencias de Costos de Rutas Factibles de eil76.

	COSTO DE RUTA FACTIBLE
Z Opt	538
Ts-TSP UPB	580
CnW	693

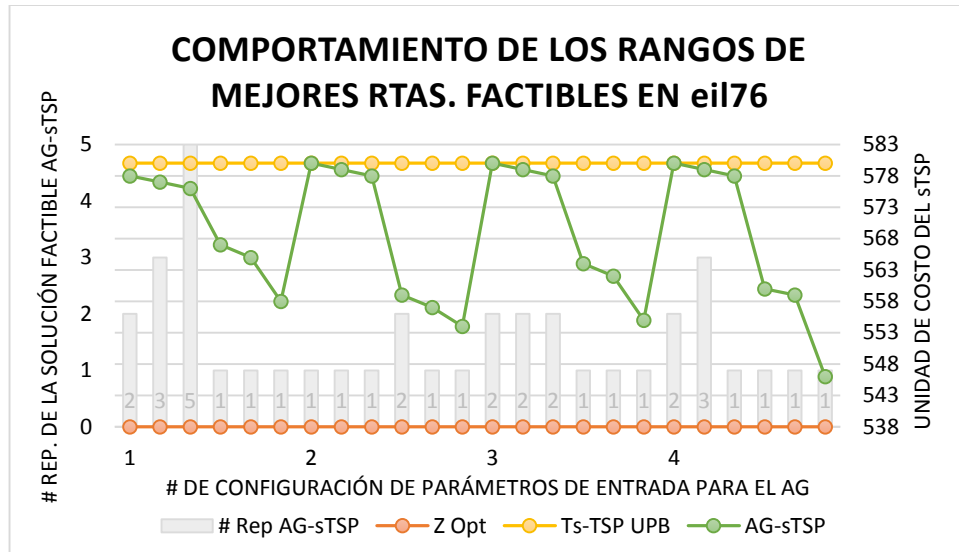
Fuente: Elaborado por el Autor.

Tabla 13. Rangos de Resultados Factibles de cada Sesión en la Instancia eil76.

PARÁMETROS	AG-sTSP	# Rep.	Prop AG-sTSP	Z Opt Gap	Ts-TSP Gap	CnW Gap
Configuración 1	578	2	3,13%	-7,43%	0,34%	16,59%
	577	3	4,69%	-7,25%	0,52%	16,74%
	576	5	7,81%	-7,06%	0,69%	16,88%
	567	1	1,56%	-5,39%	2,24%	18,18%
	565	1	1,56%	-5,02%	2,59%	18,47%
	558	1	1,56%	-3,72%	3,79%	19,48%
Configuración 2	580	1	1,56%	-7,81%	0,00%	16,31%
	579	1	1,56%	-7,62%	0,17%	16,45%
	578	1	1,56%	-7,43%	0,34%	16,59%
	559	2	3,13%	-3,90%	3,62%	19,34%
	557	1	1,56%	-3,53%	3,97%	19,62%
	554	1	1,56%	-2,97%	4,48%	20,06%
Configuración 3	580	2	3,13%	-7,81%	0,00%	16,31%
	579	2	3,13%	-7,62%	0,17%	16,45%
	578	2	3,13%	-7,43%	0,34%	16,59%
	564	1	1,56%	-4,83%	2,76%	18,61%
	562	1	1,56%	-4,46%	3,10%	18,90%
	555	1	1,56%	-3,16%	4,31%	19,91%
Configuración 4	580	2	3,13%	-7,81%	0,00%	16,31%
	579	3	4,69%	-7,62%	0,17%	16,45%
	578	1	1,56%	-7,43%	0,34%	16,59%
	560	1	1,56%	-4,09%	3,45%	19,19%
	559	1	1,56%	-3,90%	3,62%	19,34%
	546	1	1,56%	-1,49%	5,86%	21,21%

Fuente: Elaborado por el Autor.

Gráfica 4. Comportamiento de los Rangos de Resultados Factibles en ei76.



Fuente: Elaborado por el Autor.

### 7.3.2.5 Instancia pr76.

Tabla 14. Referencias de Costos de Rutas Factibles de pr76.

COSTO DE RUTA FACTIBLE	
Z Opt	108159
Ts-TSP UPB	125799
CnW	151555

Fuente: Elaborado por el Autor.

Tabla 15. Rangos de Resultados Factibles de cada Sesión en la Instancia pr76.

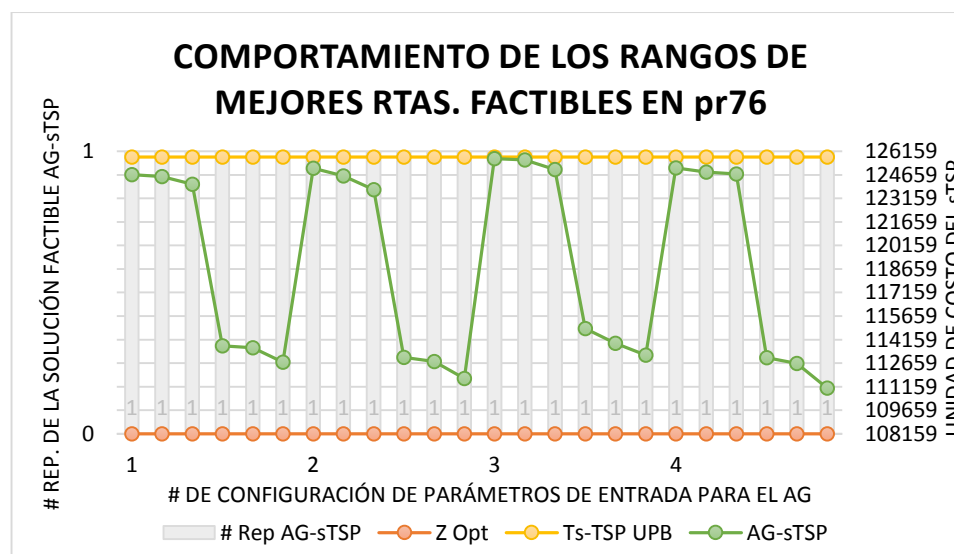
PARÁMETROS	AG-sTSP	# Rep.	Prop AG-sTSP	Z Opt Gap	Ts-TSP Gap	CnW
Configuración 1	124670	1	1,56%	-15,27%	0,90%	17,74%
	124545	1	1,56%	-15,15%	1,00%	17,82%
	124070	1	1,56%	-14,71%	1,37%	18,14%
	113757	1	1,56%	-5,18%	9,57%	24,94%
	113641	1	1,56%	-5,07%	9,66%	25,02%
	112715	1	1,56%	-4,21%	10,40%	25,63%
Configuración 2	125075	1	1,56%	-15,64%	0,58%	17,47%
	124591	1	1,56%	-15,19%	0,96%	17,79%
	123714	1	1,56%	-14,38%	1,66%	18,37%
	113031	1	1,56%	-4,50%	10,15%	25,42%
	112753	1	1,56%	-4,25%	10,37%	25,60%

Tabla 15. Rangos de Resultados Factibles de cada Sesión en la Instancia pr76. (Continuación).

<b>Configuración 2</b>	111673	1	1,56%	-3,25%	11,23%	26,32%
<b>Configuración 3</b>	125690	1	1,56%	-16,21%	0,09%	17,07%
	125619	1	1,56%	-16,14%	0,14%	17,11%
	124998	1	1,56%	-15,57%	0,64%	17,52%
	114857	1	1,56%	-6,19%	8,70%	24,21%
	113920	1	1,56%	-5,33%	9,44%	24,83%
	113167	1	1,56%	-4,63%	10,04%	25,33%
<b>Configuración 4</b>	125110	1	1,56%	-15,67%	0,55%	17,45%
	124834	1	1,56%	-15,42%	0,77%	17,63%
	124726	1	1,56%	-15,32%	0,85%	17,70%
	113009	1	1,56%	-4,48%	10,17%	25,43%
	112648	1	1,56%	-4,15%	10,45%	25,67%
	111060	1	1,56%	-2,68%	11,72%	26,72%

Fuente: Elaborado por el Autor.

Gráfica 5. Comportamiento de los Rangos de Resultados Factibles en pr76.



Fuente: Elaborado por el Autor.

**7.3.3 Análisis General de Resultados del Procesamiento AG-sTSP.** Retomando que sobre la base de las mejores soluciones factibles de las 4 últimas sesiones experimentales de cada Instancia procesada por el AG propuesto (AG-sTSP):

- ❖ Inicialmente puede observarse que en la mayoría de los casos (13 de 20), el primer costo de ruta factible hallado por el AG-sTSP en cada sesión para cada instancia, es

igual a los costos de ruta factible que proponen los resultados de Ts-TSP UPB, esto es, cuando Ts-TSP Gap es igual a 0%.

- ❖ En el caso anterior cuando Ts-TSP Gap es igual 0%, Z Opt Gap revela porcentualmente la inferioridad en términos de calidad, de las soluciones factibles halladas en la investigación Ts-TSP UPB.
- ❖ Para la instancia bays29, aunque en las tres últimas sesiones experimentales se alcanza el valor óptimo expuesto por la literatura (2,020), se puede observar que las configuraciones 2 y 4 proveen al AG-sTSP de una mayor capacidad para encontrar soluciones de mejor calidad a la presentada por Ts-TSP UPB, donde:
  1. La configuración 2 ejerce sobre el AG-sTSP una influencia moderada pero deseable, ya que se encuentra nueve veces el costo de ruta 2.026 (costo Ts-TSP UPB), cuatro veces el costo de ruta 2.022 y dos veces el costo de ruta 2.020 (costo TSPLIB).
  2. En cuanto a la configuración 4, se podría describir también un efecto deseable pero mucho más agresivo sobre el AG-sTSP, ya que se encuentra dieciocho veces el costo de ruta 2.026 (costo Ts-TSP UPB), una vez el costo de ruta 2.022 y cinco veces el costo de ruta 2.020 (costo TSPLIB).
- ❖ Se puede inferir que a medida que se contemplen instancias con un mayor número de ubicaciones, o puntos de distribución, o ciudades, etc. Para su procesamiento mediante el AG-sTSP; el valor de Prop AG-sTSP tenderá a ser el menor valor de proporción posible para un costo de ruta por sesión ( $1/64 = 1,56\%$ ) según la estrategia 2<sup>k</sup> replicado, lo anterior, de acuerdo a que el universo de posibles soluciones al sTSP sigue un comportamiento exponencial como ha sido justificado anteriormente en el Inciso 5.3.2.
- ❖ Para la instancia eil51, se puede notar como la calidad de los resultados de las dos primeras sesiones difieren de las dos últimas en una forma positiva, tal vez, debido a la dotación en el AG-sTSP de más oportunidades para encontrar respuestas de mayor fitness y entre las dos últimas sesiones, tal vez debido a una mayor pero adecuada retención de la información genética del individuo elite para la mejora de la población de tratamiento.
- ❖ Para la instancia eil51, se puede evidenciar un estrago en la calidad de las soluciones factibles entre la sesión 1 y 2 (es decir, que no siguen la tendencia de mejorar, descrita entre las sesiones 3 y 4), tal vez, causado por una desproporcionalidad entre los parámetros de entrada “*Tenencia\_Elite*” y “*Criterio\_Parada\_AG*”, donde “*Tenencia\_Elite*” asume un valor de gran magnitud que resulta difícil de asimilar por “*Criterio\_Parada\_AG*”.

- ❖ Según los resultados del procesamiento de la instancia st70, se puede observar una aparente idoneidad de la configuración 2 en el AG-sTSP en cuanto al alcance de soluciones de calidad ligeramente mejores a las obtenidas mediante la configuración 4 (mejor Z Opt Gap de la sesión 2 = -3.26%, mejor Z Opt Gap de la sesión 4 = -3.41%), aunque esta última, presente características más atractivas de Prop AG-sTSP que en la configuración 2.
- ❖ Como puede verse entre las Gráficas 4 y 5, para este tipo de instancias el AG-sTSP posee la capacidad de alcanzar respuestas aparentemente atípicas de excelente calidad, es decir, que no siguen un comportamiento linealmente esperado respecto al costo de ruta óptimo expuesto por la literatura “Z Opt”, lo anterior puede verse evidenciado en el cambio abrupto de pendiente entre el mejor costo de ruta factible de cada configuración y aquel costo de calidad inmediatamente inferior.

**7.3.4 Diseño de Experimentos (DOE) de Efectos Fijos para la Instancia bays29.** En el presente apartado, como parte de la oferta de valor de la presente investigación aplicada de pregrado, mediante la dinámica que enmarca el algoritmo secuencial de ramificación y limitado (Branch-and-Bound, B&B), se permite recomendar una posiblemente buena configuración de parámetros de entrada del AG-sTSP para el procesamiento replicable de la instancia sTSP-TSPLIB “bays29”; dicha elección es justificada de acuerdo a que esta, por razones de practicidad en la aplicación del B&B, comprende el costo computacional más bajo en relación a las otras cuatro instancias tomadas en la presente investigación para efectos de experimentación con el AG-sTSP.

La dinámica y el resultado de la aplicación de la metodología B&B para la emisión de dicha recomendación, puede representarse claramente con ayuda de la ilustración 7, donde:

- ❖ Cada nivel del árbol representa la consideración de uno de los diferentes factores codificados en el AG-sTSP (Ver Tabla 3), además, cabe resaltar que no se tiene en cuenta un orden o característica específica para la representación de los factores codificados por parte de los niveles,
- ❖ Los nodos contenidos en cada nivel, describen una variante de dicho factor para su experimentación con el AG-sTSP bajo el enfoque B&B,
- ❖ Se tiene en cuenta los siguientes 4 “predictores B&B” propuestos (valores en los que el algoritmo B&B se basa para permitir la expansión o limitación del árbol de búsqueda de soluciones):
  - 1. Puntaje del Arreglo.** Corresponde al conteo de respuestas factibles del experimento elemental a lo largo de sus réplicas, cuando el costo de ruta encontrado es menor o igual al costo Ts-TSP UPB para bays29.

**2. Puntaje de Diferencia.** Corresponde a la suma de las diferencias del costo óptimo de la literatura “Z Opt” frente a cada uno de los costos factibles encontrados en las réplicas que comprenden cada uno de los experimentos elementales.

$$3. \quad \% \text{ Mejores RTAs} = \left( \frac{\text{Num.Mejores.Rtas}}{\text{Núm.Buenas.Rtas}} \right) \quad (5)$$

$$4. \quad \% \text{ Buenas RTAs} = \left( \frac{\text{Núm.Buenas.Rtas}}{\text{Núm.Exp.Sesión}} \right) \quad (6)$$

Donde para 3 Y 4:

**Num. Mejores. Rtas** = Núm. de respuestas de la sesión, que son de mejor calidad que las encontradas por Ts-TSP UPB.

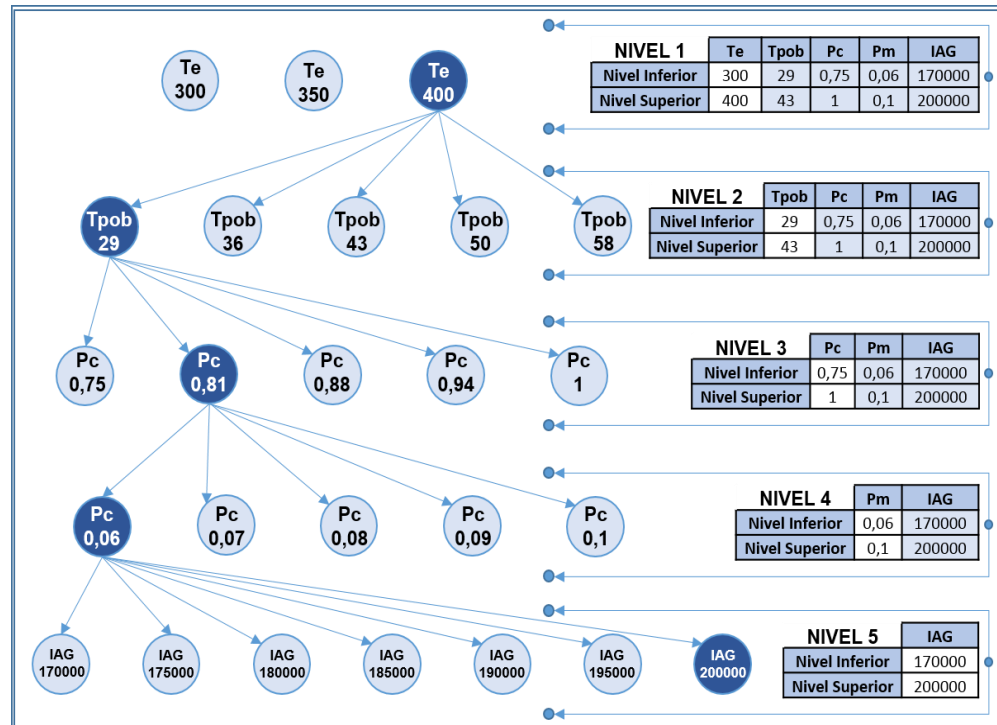
**Núm. Buenas. Rtas** = Núm. de respuestas de la sesión, que son de igual y/o mejor calidad que las encontradas por Ts-TSP UPB.

- ❖ Los predictores B&B propuestos presentan una jerarquía de relevancia en un orden descendente; dicha jerarquía, hace descartable la medida de los otros predictores B&B inmediatamente inferiores cuando el predictor B&B que se compara entre los nodos factibles del árbol son diferentes entre sí, y
- ❖ Se tienen en cuenta que entre más positivo sea el valor del predictor B&B en evaluación, mayor es el atractivo del nodo para permitir su posterior expansión.

Para la implementación de la metodología B&B, se tomó como nodo de referencia, aquella sesión de bays29 que en su etapa experimental primaria alcanzó los mejores resultados en el desarrollo de los experimentos elementales  $2^k$  replicado, por tanto, la 4ta sesión de bays29 en su etapa experimental primaria corresponde a el nodo **Te = 300**.

### 7.3.5 Análisis de Resultados DOE Efectos Fijos Según B&B para la Instancia bays29.

Ilustración 7. Árbol B&B para la Configuración de Parámetros de Entrada en bays29.



Fuente: Elaborado por el Autor

En el nivel 1, el mayor *Puntaje del Arreglo* fue de 3 para los nodos Te 300, Te 350 y Te 400. Al revisar el *Puntaje de Diferencia*, se encontraron respectivamente los puntajes -72, -82 y -22, concluyendo así, que para el nivel 1 el nodo **Te 400** es quien mejor representaría al parámetro de entrada *Tenencia\_Elite*.

En el nivel 2, el mayor *Puntaje del Arreglo* fue de 3 para los nodos Tpob 29, Tpob 50 y Tpob 58. Luego, al revisar el *Puntaje de Diferencia*, el valor más positivo que se encontró fue de -22 para los nodos Tpob 29 y Tpob 58. Luego al revisar el *% Mejores RTAs*, se encontró en el nodo Tpob 29 un valor de 46,154%, mientras que para el nodo Tpob 58 se encontró un valor de 27,778%.

Aunque como conclusión para el nivel 2, el nodo **Tpob 29** es quien mejor representaría al parámetro de entrada *Tam\_Pob*. Se hizo un ejercicio adicional que comprobaría la dinámica B&B teniendo en cuenta que el valor del predictor *% Buenas RTAs* en el nodo Tpob 58 era mayor que en el nodo Tpob 29 (56,25% y 40,625% respectivamente).

Entonces, en el nivel 3, mediante Tpob 29, se encontró como mayor *Puntaje del Arreglo* el valor 4 con un *Puntaje de Diferencia* de -18 en el nodo Pc 0,81; mientras que mediante

Tpob 58, se encontró como mayor *Puntaje del Arreglo* el valor de 3 con un *Puntaje de Diferencia de* -22 en el nodo Pc 0,75; por tanto, aquí se puede evidenciar la razón por la cual, de acuerdo a la dinámica B&B propuesta basada en los predictores, el nodo Tprob 58 fue limitado y el nodo Tprob 29 fue adecuadamente expandido.

Según lo anterior, para el nivel 3, se permitió concluir que el nodo **Pc 0,81** es quien mejor representaría al parámetro de entrada *Prob\_Cruce*.

En el Nivel 4, el mayor *Puntaje del Arreglo* encontrado hasta el momento fue de 4 en el nodo Pm 0,06, concluyendo así, que para el nivel 4 el nodo **Pm 0,06** es quien mejor representaría al parámetro de entrada *Prob\_Mutacion*.

En el Nivel 5, el mayor *Puntaje del Arreglo* encontrado hasta el momento fue de 4 en el nodo IAG 200000, concluyendo así, que para el nivel 5 el nodo **IAG 20000** es quien mejor representaría al parámetro de entrada *Criterio\_Parada\_AG*.

Remítase a la carpeta de Anexo “00 Tabulación de Resultados AG para Instancias sTSP-TSPLIB”, el archivo 05 de extensión “.xlsx”, para una consulta completa de los resultados de los experimentos elementales  $2^k$  replicado de la instancia bays29 bajo el enfoque B&B.

### 7.3.6 Recomendación DOE Efectos Fijos Según B&B para la Instancia bays29.

De acuerdo al inciso 7.3.5, a continuación en la tabla 16, se emite la recomendación de una posiblemente buena configuración de parámetros de entrada del AG-sTSP para el procesamiento replicable de la instancia sTSP-TSPLIB “bays29”.

**Tabla 16. Configuración Recomendada de los Parámetros de Entrada AG-sTSP para bays29.**

PARÁMETRO DE ENTRADA AG-sTSP		VALOR
NOMBRE	CODIFICACIÓN	
Tamaño de la Población de Tratamiento	<i>Tam_Pob</i>	29
Probabilidad de Cruce de Individuos	<i>Prob_Cruce</i>	0,81
Probabilidad de Mutación del Individuo	<i>Prob_Mutacion</i>	0,06
Número de Iteraciones para el AG	<i>Criterio_Parada_AG</i>	200.000
Tenencia Elite	<i>Tenencia_Elite</i>	400

Fuente: Elaborado por el Autor.

**7.3.7 Tiempos de Procesamiento Promedio de Instancias sTSP-TSPLIB.** A continuación se presentan en las siguientes tablas del número 17 al 23, los tiempos promedios de solución mediante la implementación de las metaheurísticas de solución TS-sTSP UPB (Un Enfoque de Búsqueda Tabú para el TSP por Jhon G. G. Becerra) y AG-sTSP (Presente investigación aplicada de pregrado) para posterior comparación.

Tabla 17. Tiempo Promedio de Procesamiento de Instancias TSPLIB con Ts-TSP UPB.

INSTANCIAS	Ts-TSP UPB (s)	Ts-TSP UPB (min)
bays29	155	2,583333333
eil51	3015	50,25
st70	19064	317,7333333
eil76	21634	360,5666667
pr76	23416	390,2666667

Fuente: Elaborado por el Autor.

Tabla 18. Tiempo Promedio de Procesamiento AG-sTSP de bays29.

	AG-sTSP bays29	
	IAG = 200000	IAG = 170000
n	172	172
Promedio (min)	6,16637597	5,26782946
Desviación Est.	1,24494361	1,06326147

Fuente: Elaborado por el Autor.

Tabla 19. Tiempo Promedio de Procesamiento AG-sTSP de eil51 Parte 1.

	AG-sTSP	
	IAG = 180000	IAG = 150000
n	64	64
Promedio (min)	15,73040541	13,14900334
Desviación Est.	2,439916956	2,041682817

Fuente: Elaborado por el Autor.

Tabla 20. Tiempo Promedio de Procesamiento AG-sTSP de eil51 Parte 2.

	AG-sTSP eil51	
	IAG = 300000	IAG = 250000
n	64	64
Promedio (min)	25,26689486	21,32643316
Desviación Est.	4,573887481	3,293311752

Fuente: Elaborado por el Autor.

Tabla 21. Tiempo Promedio de Procesamiento AG-sTSP de st70.

	AG-sTSP st70	
	IAG = 500000	IAG = 350000
<b>n</b>	128	128
<b>Promedio (min)</b>	83,49016091	59,19007645
<b>Desviación Est.</b>	15,37331667	10,90758131

Fuente: Elaborado por el Autor.

Tabla 22. Tiempo Promedio de Procesamiento AG-sTSP de eil76.

	AG-sTSP eil76	
	IAG = 500000	IAG = 350000
<b>n</b>	128	128
<b>Promedio (min)</b>	104,6022135	73,99518229
<b>Desviación Est.</b>	18,56105427	12,95436707

Fuente: Elaborado por el Autor.

Tabla 23. Tiempo Promedio de Procesamiento AG-sTSP de pr76.

	AG-STSP pr76	
	IAG = 500000	IAG = 350000
<b>n</b>	128	128
<b>Promedio (min)</b>	105,2809896	74,03645833
<b>Desviación Est.</b>	18,32962773	12,53488357

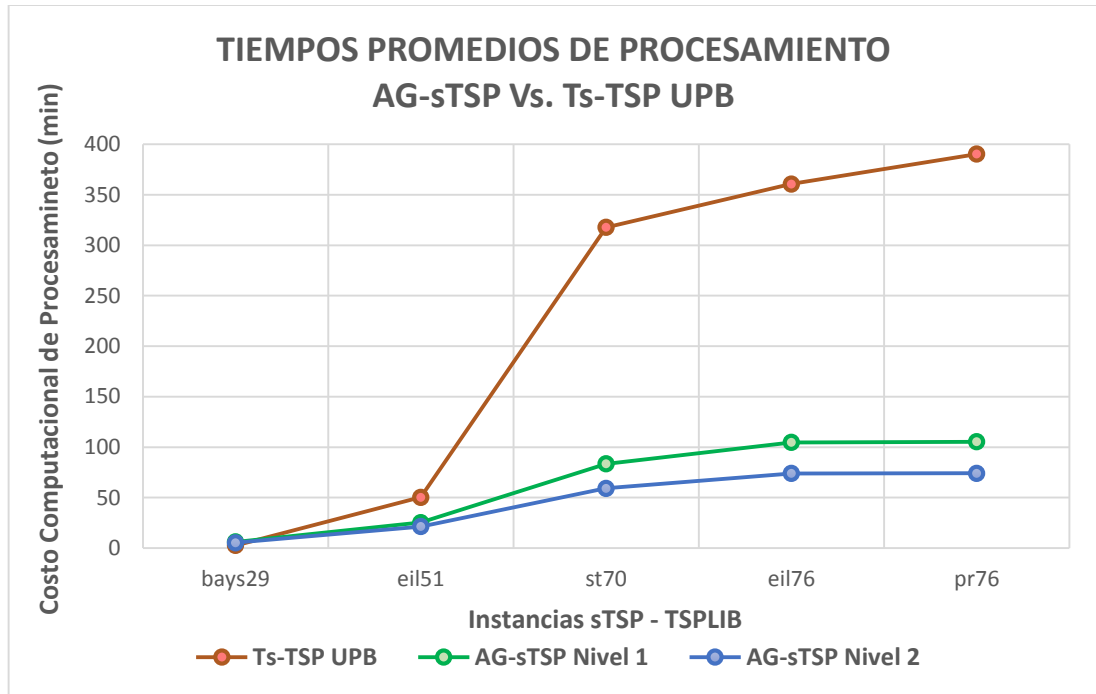
Fuente: Elaborado por el Autor.

**7.3.8 Tiempos de Procesamiento Promedio Ts-TSP UPB Vs. AG-sTSP.** En la siguiente Gráfica 6, se puede observar claramente la diferencia en términos de tiempo computacional de procesamiento de las instancias sTSP-TSPLIB consideradas, entre las metaheurísticas de solución “Búsqueda Tabú” y “Algoritmo genético”.

**Consideraciones:**

- ❖ Para la instancia eil51 se tuvo en cuenta los dos niveles de los tiempos promedios de procesamiento de la parte 2 (Ver Tabla 20), ya que las configuraciones con **IAG = 300.000** e **IAG = 250.000**, presentaron respuestas de mejor calidad en comparación a la parte 1.
- ❖ En la investigación Ts-TSP UPB no se especifica característica alguna del hardware del(os) ordenador(es) utilizado(s) para la implementación de dicha propuesta en la solución de las instancias sTSP-TSPLIB consideradas.

Gráfica 6. Tiempos Promedios de Procesamiento AG-sTSP Vs. Ts-TSP UPB.



Fuente: Elaborado por el Autor.

#### 7.4 IMPLEMENTACIÓN DEL MÉTODO DE SOLUCIÓN EXACTA (MSE) EN LA RESOLUCIÓN DEL sTSP

Para el desarrollo de este apartado se hizo uso del software GAMS (General Algebraic Modeling System), el cual, es considerado como un sistema de modelado de alto nivel para la programación matemática y optimización, diseñado específicamente para el modelado lineal, problemas de optimización no lineal entera y mixta<sup>43</sup>. Por tanto se justifica su utilización debido a que este, es especialmente útil en el tratamiento de problemas grandes y complejos como el sTSP.

Para efectos de comparación con el AG-sTSP: Adicionalmente a los modelos de programación lineal que se formulan en la presente investigación aplicada de pregrado para la solución de cada instancia TSPLIB considerada (Método de Solución Exacta - MSE); se hizo uso del modelo de programación lineal planteado en la documentación contribuida del dominio oficial de GAMS titulada "Mathematical Optimization Models"<sup>44</sup> (MOM).

<sup>43</sup> GAMS. General Algebraic Modeling System. [On Line] URL: <<http://www.gams.com/docs/intro.htm>>

<sup>44</sup> Linares, Pedro; Ramos, Andrés; Sánchez, Pedro; Sarabia, Ángel, Vitoriano, Begoña. Modelos de Optimización Matemática. [En Línea] Disponible en Pdf.

#### 7.4.1 Tiempo Promedio de Procesamiento de Instancias sTSP-TSPLIB con MSE.

Tabla 24. Tiempos Promedios de Procesamiento de instancias TSPLIB con MSE y MOM.

INSTANCIA	NÚM. RESTRICCIONES MSE	TIEMPO DE PROCESO MSE (h:m:s.ms)	TIEMPO DE PROCESO MOM (h:m:s.ms)
bays29	42	00:00:00.463	00:16:52.599
		00:00:00.457	00:16:51.972
		00:00:00.474	00:16:51.763
		00:00:00.454	00:16:51.855
		00:00:00.451	00:16:51.720
		00:00:00.459	00:16:52.192
<b>TIEMPO PROM. DE PROCESO MSE</b>		<b>00:00:00.460</b>	<b>00:16:52.017</b>
eil51	42	00:00:00.458	Cap. de Memoria Excedida
		00:00:00.456	Cap. de Memoria Excedida
		00:00:00.458	Cap. de Memoria Excedida
		00:00:00.460	Cap. de Memoria Excedida
		00:00:00.444	Cap. de Memoria Excedida
		00:00:00.461	Cap. de Memoria Excedida
<b>TIEMPO PROM. DE PROCESO MSE</b>		<b>00:00:00.456</b>	<b>- Indeterminado -</b>
st70	183	00:00:05.075	Cap. de Memoria Excedida
		00:00:04.967	Cap. de Memoria Excedida
		00:00:04.950	Cap. de Memoria Excedida
		00:00:04.972	Cap. de Memoria Excedida
		00:00:04.972	Cap. de Memoria Excedida
		00:00:04.907	Cap. de Memoria Excedida
<b>TIEMPO PROM. DE PROCESO MSE</b>		<b>00:00:04.974</b>	<b>- Indeterminado -</b>
eil76	85	00:00:00.644	Cap. de Memoria Excedida
		00:00:00.641	Cap. de Memoria Excedida
		00:00:00.642	Cap. de Memoria Excedida
		00:00:00.645	Cap. de Memoria Excedida
		00:00:00.646	Cap. de Memoria Excedida
		00:00:00.643	Cap. de Memoria Excedida
<b>TIEMPO PROM. DE PROCESO MSE</b>		<b>00:00:00.643</b>	<b>- Indeterminado -</b>
pr76	204	00:00:52.571	Cap. de Memoria Excedida
		00:00:52.316	Cap. de Memoria Excedida
		00:00:52.439	Cap. de Memoria Excedida

<[http://www.gams.com/docs/contributed/modelado\\_en\\_gams.pdf](http://www.gams.com/docs/contributed/modelado_en_gams.pdf)> Universidad Pontificia Comillas, Escuela Técnica Superior de Ingenierías. p. 27-29. [Consultado el 22 de Julio de 2014]

**Tabla 24. Tiempos Promedios de Procesamiento de instancias  
TSPLIB con MSE y MOM. (Continuación)**

<b>pr76</b>	<b>204</b>	00:00:52.565	Cap. de Memoria Excedida
		00:00:52.643	Cap. de Memoria Excedida
		00:00:52.750	Cap. de Memoria Excedida
<b>TIEMPO PROM. DE PROCESO MSE</b>		<b>00:00:52.547</b>	<b>- Indeterminado -</b>

Fuente: Elaborado por el Autor.

**7.4.2 Análisis General de Resultados del Procesamiento MSE.** Debido a que la soluciones halladas mediante MSE para cada instancia considerada (en términos de costo de ruta factible) son iguales a los costos de ruta óptima “Z Opt” expuestas por TSPLIB, el siguiente análisis de resultados para la evaluación del desempeño computacional del AG-sTSP, se centra en la consideración del número necesario de restricciones MSE y tiempos promedios de procesamiento de dichas instancias.

- ❖ Se puede inferir que existe una fuerte tendencia, de producto exponencial, entre el Número de Restricciones MSE necesarias para el procesamiento de las instancias TSPLIB consideradas y los resultados “Tiempo de Proceso MSE”.
- ❖ Al inicio de la experimentación con el MSE, se esperaba que se presentara con la instancias consideradas, el mismo comportamiento anteriormente mencionado, entre la aparente complejidad de solución de la instancia (número de ubicaciones, o puntos de distribución, o ciudades, etc. Que contiene la instancia) y el Número de Restricciones MSE; lo cual, resultó ser una especulación invalidada por los casos presentados con las instancias eil51 y eil76.
- ❖ Como se puede observar en el caso de la instancia bay29, la diferencia entre los tiempos de solución MSE y MOM es supremamente exagerada, siendo MOM el caso más cercano a la utilización de una metaheurística de solución pero desde el ámbito de la programación lineal del problema de optimización.
- ❖ A grandes rasgos se plantea la idea de que el tiempo de solución de las instancias utilizando MOM, aumenta exponencialmente conforme aumenta el número de ubicaciones, o puntos de distribución, o ciudades, etc. que contenga la instancia, es decir, el tiempo MOM aumenta exponencialmente de acuerdo a la complejidad matricial que presente la instancia TSPLIB en consideración.

Para el caso de la instancia TSPLIB bays29, la programación MOM muestra una considerable inferioridad respecto al desempeño del AG-sTSP en términos de tiempo de solución, pues cuando se utiliza la configuración de parámetros de entrada que comprende un IAG de 170.000, el Gap del AG-sTSP respecto a MOM (Gap MOM) es de 68.77% y cuando se comprende un IAG de 200.000 el Gap MOM es de 63,44%.

## LIMITACIONES

- ❖ A pesar de que la codificación del AG-sTSP para su implementación en la solución del sTSP, se desarrolló en el entorno de programación que ofrece “Visual Basic para aplicaciones v.7.1” como extensión del software Microsoft Excel. En la fase del diseño del AG se procuró la utilización de una metodología de Programación Orientada Objetos (POO) que tuvo como fin principal, la interrelación de cada uno de los procedimientos heurísticos que comprenden las etapas de dicho Algoritmo Evolutivo (Ver inciso 7.1), lo anterior, asegurando la adecuada comunicación, manipulación y gestión interna de la información para el procesamiento de la instancia considerada.
- ❖ Aunque no figure en el alcance de la presente investigación aplicada de pregrado, el análisis de resultados bajo el enfoque de la estrategia experimental  $2^K$  replicado que tenía como fin principal la emisión de configuraciones factibles de los parámetros de entrada del AG-sTSP para el procesamiento eficiente de las instancias consideradas, no fue posible: 1. Debido a la violación de los supuestos de independencia muestral, comportamiento normal y homocedasticidad de los datos; 2. Desconocimiento de técnicas especializadas para la transformación de los datos y 3. Limitación del tiempo disponible para conclusión de la investigación.
- ❖ Debido a que en la investigación Ts-TSP UPB no se especifica característica alguna del hardware del(os) ordenador(es) utilizado(s) para la implementación de dicha propuesta en la solución de las instancias sTSP-TSPLIB consideradas. La apreciación Ts-TSP UPB Vs. AG-sTSP adquiriría un valor mucho más cualitativo que cuantitativo, donde a grandes rasgos se podría considerar la superioridad de la técnica AG-sTSP para encontrar “soluciones de calidad buena” (soluciones de calidad igual y/o mejor a la calidad de las soluciones encontradas por Ts-TSP UPB) con menores costos de tiempo computacional promedio, a excepción del caso bays29.
- ❖ No se pudo obtener los tiempos promedios de procesamiento MOM de las instancias eil51, st70, eil76 y pr76 debido a que muy seguramente, la complejidad matricial que presentan dichas instancias, superan la capacidad máxima de memoria que dispone el ordenador para su procesamiento en el software GAMS (Ver Tabla 24).

## CONCLUSIONES

- ❖ Se identificó en la dinámica que comprende el seudocódigo de un AG, una serie de aspectos relevantes que de acuerdo a su flexibilidad, permitieron la estructuración de una propuesta de empalme de conceptos ajenos a los que predeterminadamente hace uso dicha metaheurística de solución.
- ❖ Se logró materializar la propuesta del diseño de un Algoritmo Genético que sobre la base que presenta el funcionamiento de su seudocódigo, fue internamente modificado y utilizado en la búsqueda efectiva de soluciones factibles para 5 diferentes instancias sTSP-TSPLIB.
- ❖ Se evidenció la sobresaliente capacidad del AG-sTSP frente al Ts-TSP UPB, de presentar una mayor efectividad de procesamiento de las instancias TSPLIB consideradas para encontrar soluciones factibles de mejor calidad con menores costos de tiempo computacional (en cuanto a tiempo computacional, se describió una excepción en el procesamiento de bays29).
- ❖ Se diseñaron e implementaron en GAMS, los modelos de optimización lineal (MSEs) que permitieron esbozar la magnitud del problema sTSP, de acuerdo a la requisición del número de restricciones necesarias para la eliminación de subciclos en el establecimiento de las rutas “óptimas” de cada instancia en consideración.
- ❖ Se implementó el concepto de individuo semilla o ancestro común en el AG-sTSP, a partir de la solución presentada por el algoritmo de ahorros de *Clarke & Wright*, para conformación de la población inicial de tratamiento mediante el proceso de mutación BIT-SPLIT, lo anterior, corroboró la gran aplicación de la recomendación: “las soluciones iniciales buenas, usualmente producen buenos resultados con menor costo de tiempo”. [37; NASEEM J, YANG]
- ❖ Se incorporó de manera cuidadosa en el AG-sTSP el concepto de población elite, el cual hizo notable, la propia dotación de una poderosa capacidad de exploración para encontrar soluciones de buena calidad en una región determinada del espacio de soluciones, región la cual, inicialmente fue conformada por la mutación sucesiva de la información genética del ancestro común. Lo anterior, se soportó en la conclusión: “El elitismo mejora la búsqueda local a expensas de la perspectiva global”. [19; DE JONG]
- ❖ Se Instauró en el AG-sTSP el concepto de Población Inadaptada que tuvo un efecto de contribución significativa en el control y mejoramiento de la adaptabilidad promedio de la poblaciones concurrentes de soluciones, proveyendo al AG-sTSP la capacidad de aprovechar la bondad que presenta la convergencia metaheurística de largo plazo. Lo anterior solo puede ser visible en el resultado del procesamiento de la instancia mediante el AG-sTSP, en la columna llamada “ADAPT. MEDIA”.

Remítase al Anexo G, para la consulta gráfica que evidencia la conclusión N° 7, donde la pendiente de la línea de tendencia toma un sentido negativo ( $m = 8 \times 10^{-4}$ ).

- ❖ Se identificó la existencia de una relación positivamente proporcional entre el Número de Restricciones MSE y el Tiempo Promedio de Procesamiento MSE, en este caso, no se puntualizó una relación de proporción directa debido a que los Tiempos de Procesamiento MSE describen un comportamiento exponencial conforme aumenta el Número de Restricciones MSE necesarias para la eliminación de subciclos de la ruta factible.
- ❖ Se observó a lo largo del desarrollo de los experimentos elementales que el parámetro de entrada *Tenencia\_Elite*, es quien probablemente, tiene mayor influencia en la estabilización del comportamiento del AG-sTSP en la búsqueda de soluciones factibles al sTSP, permitiendo así, la convergencia metaheurística de la técnica con una variabilidad de solución cada vez menor. Esta es la tendencia aparente que presentan los conjuntos de costos factibles de las sesiones experimentales aplicadas a cada una de las instancias consideradas.
- ❖ Se evidenció mediante el desarrollo de las observaciones experimentales (para el caso del AG-sTSP), que la relación de los parámetros de entrada *Tenencia\_Elite* y *Criterio\_Parada\_AG*, resulta ser contraproducente para el procesamiento de las instancias si no se contempla la búsqueda de un equilibrio entre los mismos.
- ❖ Se ultimó que dos consecuencias directas de la omisión de la anterior conclusión serían: 1. que el AG-sTSP sufra un encapsulamiento en un óptimo local cuando la magnitud de *Tenencia\_Elite* es demasiado grande para ser asimilado por *Criterio\_Parada\_AG* y 2. que el AG-sTSP adquiera un comportamiento de búsqueda meramente aleatorio cuando la situación que se presenta, es caso viceversa.
- ❖ Se identificó que las estrategias de intensificación y diversificación deben actuar una en pro de la otra, con el objetivo de: 1. Dirigir la búsqueda hacia una región prometedora de soluciones de factibles y 2. Permitir el desplazamiento de la búsqueda a otro sector del espacio de soluciones.
- ❖ Se acreditó la capacidad de tratamiento del AG-sTSP, para procesar instancias TSPLIB de la naturaleza aTSP (*Asymmetric TSP*), lo preliminar, Debido a que la codificación propuesta de la presente metaheurística de solución, contempla la totalidad de la información dispuesta por la instancia en su lista o matriz de distancias.

## RECOMENDACIONES

- ❖ La inexperiencia en el campo de la utilización de metaheurísticos de solución para resolver problemas de optimización combinatoria, puede conllevar con una alta probabilidad, a la subestimación del presupuesto de recursos que se requieren para el desarrollo efectivo de este tipo de herramientas.
- ❖ Se sugiere al aspirante investigador que quiera incursionar en el campo de la utilización de metaheurísticos de solución, tener un nivel considerable de algún lenguaje de programación, que facilite el proceso de estructuración de la propuesta de investigación; ya que los pseudocódigos, casi que en su generalidad, desde la lectura describen una aparente simplicidad para efectos de su implementación.
- ❖ Se aconseja hacer una adecuada planeación del diseño de los experimentos para la emisión de resultados. Ya que sobre la base de estos, se emiten las conclusiones, tesis, supuestos, e incluso la continuidad de la investigación en pro del mejoramiento continuo del conocimiento generado.
- ❖ Se recomienda al aspirante investigador, hacer un inventario adecuado de los recursos que se requieren para el desarrollo de este tipo de investigaciones, lo anterior, con el fin de identificar el alcance, limitaciones y producto final de su efectiva realización.
- ❖ Es necesario que en la fase del diseño personalizado de una metaheurística de solución, en lo posible, se estructure un enfoque balanceado entre las estrategias de intensificación y diversificación evitando así, la convergencia prematura de la técnica a un óptimo local o la divergencia prolongada de la técnica hacia un óptimo global.
- ❖ Se recomienda realizar un estudio de efectos e interacciones de factores en los resultados DOE  $2^k$  replicado de la presente investigación aplicada de pregrado, con el fin de que se pueda emitir una configuración factible de parámetros de entrada para cada instancia considerada, que permita mejorar la efectividad de procesamiento del AG-sTSP en términos de tiempo y calidad de solución.
- ❖ Se recomienda estudiar la naturaleza de los operadores genéticos de cruce propuestos “MOX y COX”, con el fin de determinar la contribución metaheurística de cada uno de ellos en el AG-sTSP para el procesamiento eficiente de las instancias sTSP-TSPLIB y/o aTSP-TSPLIB.
- ❖ Se recomienda al aspirante investigador, tener “como mínimo” un buen nivel de comprensión y lectura del idioma inglés, pues es en este idioma que abunda la información científica disponible, indiferentemente de la nacionalidad de los autores.

- ❖ Se invita al aspirante investigador, tener y/o desarrollar disciplina, paciencia, perseverancia, curiosidad y un pensamiento positivista. Centrarse en la consecución de los logros formalmente propuestos y logros no propuestos, pues mientras el proceso de investigación aún tenga vitalidad dentro del alcance definido, “Uno se puede dar cuenta de que se da y se recibe más, de lo que inicialmente se ha presupuestado”.

## BIBLIOGRAFÍA

- [1] AHMED, Zakir H. Genetic Algorithm for the Traveling Salesman Problem using Sequential Constructive Crossover Operator. *International Journal of Biometrics & Bioinformatics (IJBB)*; January, 2010. Vol. 3, Issue 6, p. 96-105. p. 98. ISSN: 1985-2347.
- [2] AHMED, Zakir H. A Hybrid Genetic Algorithm for the Bottleneck Traveling Salesman Problem. *ACM Transactions on Embedded Computing Systems (TECS)*; January, 2013. Artículo 9. ISSN: 1539-9087 EISSN: 1558-3465.
- [3] ALBA, Enrique; LAGUNA, Manuel; MARTÍ, Rafael. Métodos evolutivos en problemas de optimización. [En Línea] Disponible en URL: <http://www.uv.es/rmarti/paper/cpapers.html> En *Ingeniería UC* 10 (3), pp. 80-89.
- [4] ARAUJO, Lourdes; CERVIGÓN, Carlos. *Algoritmos Evolutivos: Un Enfoque Práctico*. 1 ed. México: Editorial Alfaomega grupo editor S.A., Abril, 2009. p. 68. ISBN: 978-84-7897-911-0.
- [5] AVILA-DÍAZ, F. B.; CASTILLO-MONROY, E. F.; URIBE-RODRÍGUEZ, A. Application of Genetic Algorithms for Designing Cost Optimal Heat Exchanger Networks. *Latin American Applied Research*; 2008. Vol. 38, Issue 3. ISSN: 0327-0793.
- [6] BÄCK, Thomas. Optimal Mutation Rates in Genetic Search. *Proceedings of the 5th International Conference on Genetic Algorithms*; June, 1993. p. 2-8. ISBN: 1-55860-299-2.
- [7] BASNET, C. A hybrid Genetic Algorithm for a Loading Problem in Flexible Manufacturing Systems. *International Journal of Production Research*; February, 2012. Vol. 50 Issue 3, p. 707-718. ISSN: 0020-7543.
- [8] BENREJEB, M.; BORNE, Pierre; Karray, Asma. A Hybrid Method Based on Genetic Algorithms and Tabu Search for the Single-Machine Scheduling Problem in Agro-Food Industry. *International Review of Automatic Control (I.RE.A.CO.)*; March, 2011. Issue 2, Vol. 4. p. 280-287. ISSN: 1974-6059.
- [9] BHANDARI, Dinabadhhu; MURTHY, C.A.; PAL, Sankar K. Variance as a Stopping Criterion for Genetic Algorithms with Elitist Model. *Fundamenta Informaticae*;

December, 2012. Issue 3, Vol. 120. p. 145-164. ISSN: 0169-2968 (Print), 1875-8681 (Online).

- [10] BRITO SANTANA, Julio; CAMPOS RODRÍGUEZ, Clara; GARCÍA LÓPEZ C. Félix; GARCÍA TORRES, Miguel; MELIÁN BATISTA, Belén; MORENO P´EREZ, José A.; MORENO VEGA, J. Marcos; Metaheurísticas: una Revisión Actualizada. . [En Línea] Disponible en Pdf. URL: <<http://jamoreno.webs.ull.es/www/papers/paper37.pdf>> Grupo de Computación Inteligente Universidad de La Laguna. Departamento de Estadística, Investigación Operativa y Computación. Documento de Trabajo no 02/2004.
- [11] CHEN, Wei; LI, Wei; LIU, Hanmin; LUO, Wenjing; YAN. Xuesong; ZHANG, Can. Traveling Salesman Problem Using Particle Swarm Optimization Algorithm. International Journal of Computer Science Issues (IJCSI); November, 2012. Vol. 9 Issue 6, p. 264-271. ISSN: 16940784.
- [12] CHUNGUO, Wu; JINHUI, Yang; HEOW, Pueh Lee; YANCHUN, Liang. Solving Traveling Salesman Problems Using Generalized Chromosome Genetic Algorithm. Progress in Natural Science; July, 2008. Vol. 18, Issue 7. p. 887-892.
- [13] COCKBAINE OJEDA, Juan; SILVA URREA, Rubén. Perfeccionando Algoritmos Heurísticos para el Problema NP-C E-TSP. Revista chilena de ingeniería; 2013. Vol. 21, N° 2, p. 196-204.
- [14] CORTÉZ, Augusto. Ensayo: Teoría de la Complejidad Computacional y Teoría de la Computabilidad. [En Línea] Disponible en Pdf. URL: <[http://sisbib.unmsm.edu.pe/bibvirtualdata/publicaciones/risi/n1\\_2004/a14.pdf](http://sisbib.unmsm.edu.pe/bibvirtualdata/publicaciones/risi/n1_2004/a14.pdf)> Revista de Investigación de Sistemas e Informática; 2004. ISSN: 1815-0268.
- [15] DAIDA, Jason M.; STANHOPE, Stephen A. Evolutionary Programming, Optimal Mutation and Crossover Rates for a Genetic Algorithm Operating in a Dynamic Environment. Lecture Notes in Computer Science (LNCS); March, 1998. p. 693-702. Online ISBN: 978-3-540-68515-9.
- [16] DASGUPTA, Dipankar; MICHALEWICZ, Zbigniew. (Eds). Evolutionary Algorithms in Engineering Applications. Alemania: Editorial Springer-Verlag Berlín Heidelberg, 1997. p. 33. ISBN 3-540-62021-4.
- [17] DAVIS, Lawrence. Applying Adaptive Algorithms to Epistatic Domains. 9th international joint conference on Artificial intelligence. August, 1985. Vol. 1. p. 162-164. ISBN: 0-934613-02-8 978-0-934-61302-6.

- [18] DAZA, Julio Mario; MONTOYA, Jairo R.; NARDUCCI, Francesco. Resolución del Problema de Enrutamiento de Vehículos con Limitaciones de Capacidad (CVRP) Utilizando un Procedimiento Metaheurístico de dos fases. Medellín; Diciembre, 2009. ISSN: 1794-1237.
- [19] DE JONG, Kenneth Alan. An Analysis of the Behavior of a Class of Genetic Adaptive Systems. PhD Thesis, Ann Arbor, MI, USA, 1975.
- [20] DE LIMA JUNIOR, Francisco C.; DORIA NETO, Adrião D.; DE MELO, Jorge Dantas. Hybrid Metaheuristics Using Reinforcement Learning Applied to Salesman Traveling Problem. Traveling Salesman Problem, Theory and Applications; December, 2010. p. 213-236. ISBN: 978-953-307-426-9.
- [21] DÍAZ, A., GLOVER, F., GHAZIRI, H.M., GONZALEZ, J.L., LAGUNA, M, MOSCATO, P. y TSENG, F.T. (1996), Optimización Heurística y Redes Neuronales, Paraninfo, Madrid.
- [22] DIKA, Agni; MAXHUNI, Adnan; REXHEPI, Avni. Analysis of the Impact of Parameters Values on the Genetic Algorithm for TSP. IJCSI International Journal of Computer Science Issues; January, 2013. Vol. 10, Issue 1, No 3, ISSN (Print): 1694-0784 | ISSN (Online): 1694-0814.
- [23] DONG, Mei; FANGGENG, Zhao; JIANGSHENG, Sunc; SUJIAN, Li. Genetic Algorithm for the One-Commodity Pickup-and-Delivery Traveling Salesman Problem. Computers & Industrial Engineering; May, 2009. Vol. 56, Issue 4. p. 1642-1648.
- [24] FIDANOVA, Stefka, PAPRZYCKI, Marcin, ROEVA, Olympia. Influence of the Population Size on the Genetic Algorithm Performance in Case of Cultivation Process Modelling. Federated Conference on Computer Science and Information Systems (FedCSIS); September, 2013. p. 371-376. INSPEC: 13884749.
- [25] G. Dantzig, R. Fulkerson, S. Johnson. Solution of a Large-Scale Traveling-Salesman Problem [En Línea] URL: <<http://www.iro.umontreal.ca/~gendron/IFT6551/LECTURES/TSP.pdf>>.
- [26] HILLIER S., Frederick; LIEBERMAN J., Gerald. Introducción a la Investigación de Operaciones. 4 Edición en español. Impreso en México D.F.: MCGRAW-HILL INTERAMERICANA EDITORES, S.A.; 1997. 981 p. ISBN: 970-10-1022-1.
- [27] ISMKHAN, Hassan; ZAMANIFAR, Kamran. Developing Improved Greedy Crossover to Solve Symmetric Traveling Salesman Problem. IJCSI International

Journal of Computer Science Issues; July, 2012. Vol. 9, Issue 4, No 3, p. 121-126.  
ISSN (Online): 1694-0814.

- [28] J.T. Alander (1992). On Optimal Population Size of Genetic Algorithms. Proceedings CompEuro - Computer Systems and Software Engineering, 6th Annual European Computer Conference. May, 1992. p. 65-70. ISBN 0-8186-27603.
- [29] JAYALKSHMI, G. Andal; SATHI S. A Hybrid Genetic Algorithm as New Approach to Solve Traveling Salesman Problem. International of Computational Engineering Science; June, 2001. Vol. 2, No. 2, p. 339-355. ISSN: 1465-8763.
- [30] JINGUI Lu; MIN, Xie. Immune-Genetic Algorithm for Traveling Salesman Problem, Traveling Salesman Problem, Theory and Applications; December, 2010. p. 81-96. ISBN: 978-953-307-426-9.
- [31] JULSTROM A. Bryant. Very Greedy Crossover in a Genetic Algorithm for the Traveling Salesman Problem. SAC '95 Proceedings of the 1995 ACM Symposium on Applied Computing; February, 1995. p. 324-328. ISBN: 0-89791-658-1.
- [32] KARP, R. (1972). Reducibility among Combinatorial Problems: In Complexity of Computer Computations. [En Línea] Disponible en Pdf. URL: <<http://cgi.di.uoa.gr/~sgk/teaching/grad/handouts/karp.pdf>> Plenum Press, p. 85-103. New York.
- [33] KATAYAMA, Kengo; NARIHISHA, Hiroyuki. An Efficient Hybrid Genetic Algorithm for the Traveling Salesman Problem. Electronics & Communications in Japan, Part 3: Fundamental Electronic Science; February, 2001. Vol. 84 Issue 2, p. 76-83. ISSN: 1042-0967.
- [34] LINARES, Pedro; RAMOS, Andrés; SÁNCHEZ, Pedro; SARABIA, Ángel, VITORIANO, Begoña. Modelos de Optimización Matemática. [En Línea] Disponible en Pdf. <[http://www.gams.com/docs/contributed/modelado\\_en\\_gams.pdf](http://www.gams.com/docs/contributed/modelado_en_gams.pdf)> Universidad Pontificia Comillas, Escuela Técnica Superior de Ingenierías. p. 27-29.
- [35] MARTÍ, Rafa. Procedimientos Metaheurísticos en Optimización Combinatoria. [En Línea] Disponible en Pdf: <<http://www.uv.es/rmarti/paper/docs/heur1.pdf>> Universidad de Valencia, Facultad de matemáticas, departamento de estadística e investigación operativa. 60 p.

- [36] MEDINA V. Pedro Daniel; LOPEZ R. Ángela María. Análisis Crítico del Diseño Factorial 2k Sobre Casos Aplicados. *Scientia et Technica* Año XVIII, No 47, Abril de 2011. Universidad Tecnológica de Pereira. p. 101-106. ISSN 0122-1701.
- [37] NASEEM J, Sadaf; YANG, Shengxiang. A Hybrid Genetic Algorithm and Tabu Search Approach for Post Enrolment Course Timetabling. *Journal of Scheduling*; December, 2011, Vol. 14 Issue 6, p. 617-637. ISSN: 10946136.
- [38] NICOARĂ, Elena Simona. Mechanisms to Avoid the Premature Convergence of Genetic Algorithms. *Petroleum - Gas University of Ploiesti Bulletin, Mathematics - Informatics - Physics Series*; 2009, Vol. 61 Issue 1, p. 87-96. ISSN: 1224-4899.
- [39] OLIVER, I.M.; SMITH, D.J.; HOLLAND, J.R.C. A Study of Permutation Crossover Operators on the Traveling Salesman Problem. *Proceedings of the second International Conference on Genetic Algorithms*, October, 1987. p. 224-233. ISBN: 0-8058-0158-8.
- [40] PAPUTUNGAN, Irving Vitra; ZUKHRI, Zainudin. A Hybrid Optimization Algorithm Based on Genetic Algorithm and Ant Colony Optimization. *International Journal of Artificial Intelligence & Applications*; September, 2013, Vol. 4 Issue 5, p. 63-75. ISSN: 0975900X.
- [41] PINILLA GARCÍA, Vicente. *Investigación Operacional: Introducción Teórica y Aplicaciones en Administración*. 1 ed. Colombia, Bogotá D.C.: Ediciones Uniandes; 2005. p.140. ISBN: 958-695-161-8.
- [42] PONCE CRUZ, Pedro. *Inteligencia Artificial con Aplicaciones a la Ingeniería*. 1 ed. México: Editorial Alfaomega grupo editor S.A.; 2010. 348 p. ISBN 978-607-7854-83-8.
- [43] URSEM K. Rasmus. Diversity-Guided Evolutionary Algorithms. *7th International Conference Granada, Spain*; September 7-11, 2002. p. 462-471. ISBN: 978-3-540-44139-7.
- [44] ZEYAD RAMADAN, Saleem. Reducing Premature Convergence Problem in Genetic Algorithm, Application on Travel Salesman Problem; *Computer and Information Science*; December, 2013. Vol. 6, No. 1; p. 43-57. ISSN 1913-8989 E-ISSN 1913-8997.

## WEBGRAFÍA

- [45] Cambridge Journals. [On Line] URL: <<http://journals.cambridge.org>>
- [46] GAMS. General Algebraic Modeling System. [On Line] URL: <<http://www.gams.com/docs/intro.htm>>.
- [47] GAMS. Home Page. [On Line] URL: <<http://www.gams.com/>>.
- [48] GLOBETHICS. [On Line] URL: <<http://www.globethics.net/>>.
- [49] Informs.Pubsonline. [On Line] URL: <<http://pubsonline.informs.org/journal/opre>>.
- [50] Scielo, Scientific Electronic Library Online. [On Line] URL: <[www.scielo.org](http://www.scielo.org)>.
- [51] SCIENTIFIC RESEARCH open Access. [On Line] URL: <<http://www.scirp.org/journal/OpenAccess.aspx>>.
- [52] TSPLIB, Public Domain Library. Discrete and Combinatorial Optimization. [On Line] URL: <<http://www1.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/index.html>>
- [53] TSPLIB. Symmetric Traveling Salesman Problem (sTSP). Public domain Library. [On Line] URL: <<http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/tsp/>>.
- [54] Universidad de la República de Uruguay Facultad de ingeniería. Material de enseñanza: Técnicas de Computación Evolutiva. [En Línea] Disponible en Pdf. URL: <<http://www.fing.edu.uy/inco/cursos/geneticos/ae/CE.pdf>>.
- [55] University of Waterloo. History of TSP. [On Line] URL: <<http://www.math.uwaterloo.ca/tsp/history/index.html>>.
- [56] University of Waterloo. Number of Tours. [En Línea] URL: <<http://www.math.uwaterloo.ca/tsp/problem/pcb3cnt.html>>.
- [57] University of Waterloo. The Problem of TSP. [En Línea] URL: <<http://www.math.uwaterloo.ca/tsp/problem/index.html>>.

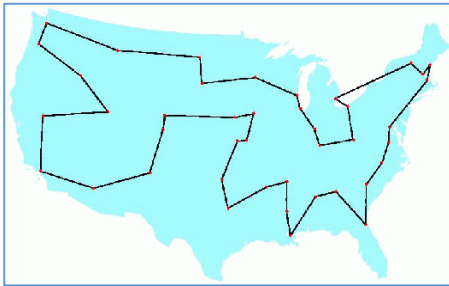
# ANEXOS

## Anexo A

### Historia Ilustrada del Avance en la Resolución del TSP.

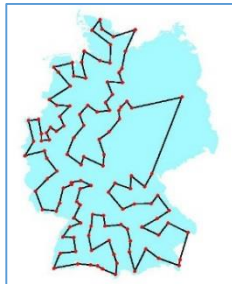
A continuación, se muestra la historia ilustrada del avance que ha presentado en cuanto a la resolución aproximada del tour óptimo para  $n$  ciudades del TSP, el cual aún hoy sigue vigente como objeto de estudio para otras técnicas de resolución especializada de problemas de optimización combinatoria:

Ilustración 8. George Dantzig, Ray Fulkerson, and Selmer Johnson (1954);  $n=49$ .



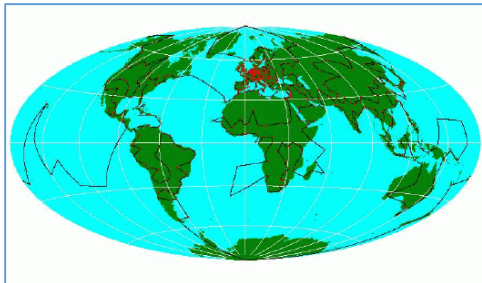
Fuente:  
<http://www.math.uwaterloo.ca/tsp/history/pictorial/dfj.html>.

Ilustración 10. Groetschel (1977);  $n=120$ .



Fuente:  
<http://www.math.uwaterloo.ca/tsp/history/pictorial/groetschel.html>.

Ilustración 12. Groetschel and Holland (1987);  $n=666$ .



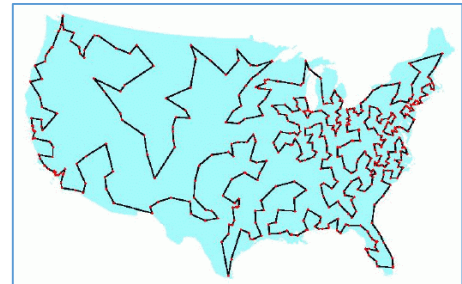
Fuente:  
<http://www.math.uwaterloo.ca/tsp/history/pictorial/gr666.html>.

Ilustración 9. Procter and Gamble Ran a Contest in 1962;  $n=33$ .



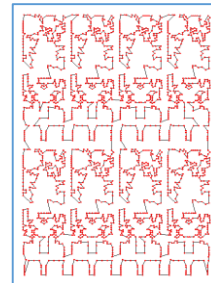
Fuente:  
<http://www.math.uwaterloo.ca/tsp/history/pictorial/car54.html>.

Ilustración 11. Padberg and Rinaldi (1987);  $n=538$ .



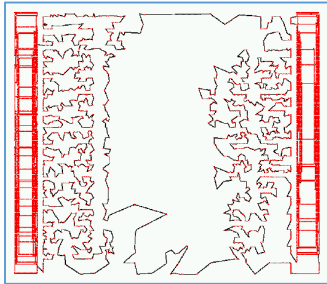
Fuente:  
<http://www.math.uwaterloo.ca/tsp/history/pictorial/att532.html>.

Ilustración 13. Padberg and Rinaldi (1987);  $n=2392$ .



Fuente:  
<http://www.math.uwaterloo.ca/tsp/history/pictorial/pr2392.html>.

Ilustración 14. Applegate, Bixby, Chvátal, and Cook (1994); n=7397.



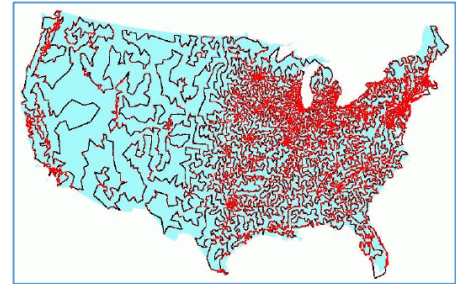
Fuente:  
<http://www.math.uwaterloo.ca/tsp/history/pictorial/pla7397.html>.

Ilustración 16. Applegate, Bixby, Chvátal, and Cook (2001); n=15112.



Fuente:  
<http://www.math.uwaterloo.ca/tsp/history/pictorial/d15112.html>.

Ilustración 15. Applegate, Bixby, Chvátal, and Cook (1998); n=13509.



Fuente:  
<http://www.math.uwaterloo.ca/tsp/history/pictorial/usa13509.html>.

Ilustración 17. Applegate, Bixby, Chvátal, Cook, and Helsgaun (2004); n=24978.



Fuente:  
<http://www.math.uwaterloo.ca/tsp/history/pictorial/sw24978.html>.

## Anexo B

**Tabla 25. Algunas Aplicaciones de la Investigación de Operaciones.**

Organización	Naturaleza de la aplicación	Año de publicación	Ahorros anuales
<b>The Netherlands Rijkswaterstatt</b>	Desarrollo de política nacional de administración del agua, incluyendo mezcla de nuevas instalaciones, procedimientos de operación y costeo.	1985	\$ 15 millones
<b>Monsanto Corp.</b>	Optimización de operaciones de producción para cumplir con metas con un costo mínimo.	1985	\$2 millones
<b>Weyerhaeuser Co.</b>	Optimización del corte de árboles en productos de madera para maximizar su producción.	1986	\$ 15 millones
<b>Electrobras/CEPAL, Brasil.</b>	Asignación óptima de recursos hidráulicos y térmicos en el sistema nacional de energía.	1986	\$ 43 millones
<b>United Airlines.</b>	Programación de turnos de trabajos en las oficinas de reservaciones y en los aeropuertos para cumplir con las necesidades del cliente a un costo mínimo.	1986	\$ 6 millones
<b>Cigto Petroleum Corp.</b>	Optimización de las operaciones de refinación y de la oferta, distribución y comercialización de productos.	1987	\$ 70 millones
<b>SANTOS, Ltd., Australia.</b>	Optimización de inversiones de capital para producir gas natural durante 25 años.	1987	\$ 3 millones
<b>San Francisco Police Department.</b>	Optimización de la programación y asignación de oficiales de patrulla con un sistema computarizado.	1989	\$ 11 millones
<b>Electric Power Research Institute.</b>	Administración de inventarios de petróleo y carbón para el servicio eléctrico con el fin de equilibrar los costos de inventario y los riesgos faltantes.	1989	\$59 millones
<b>Texaco, Inc.</b>	Optimización de la mezcla de ingredientes disponibles para que los productos de gasolina cumplieran con los requerimientos de ventas y calidad.	1989	\$ 30 millones
<b>IBM.</b>	Integración de una red nacional de inventario de refacciones para mejorar el apoyo al servicio	1990	\$ 20 millones + \$ 250 millones ahorrados en inventario
<b>Yellow Freight System, Inc.</b>	Optimización del diseño de una red nacional de transporte y la programación de rutas de envío.	1992	\$17.3 millones
<b>American Airlines.</b>	Diseño de un sistema de estructura de precios, sobreventa y coordinación de vuelos para mejorar las utilidades.	1992	\$ 500 millones más de ingresos
<b>U.S. Military Airlift Command</b>	Rapidez en la coordinación de aviones, tripulaciones, carga y pasajeros para manejar la evacuación por aire en el proyecto Tormenta del Desierto en el Medio Oriente	1992	Victoria
<b>New Haven Health Dept.</b>	Diseño de un programa efectivo de intercambio de agujas para combatir el contagio del SIDA	1993	33% menos contagios

**Fuente: Introducción a la Investigación de Operaciones.**

## Anexo C

Tabla 26. Aplicaciones de los Algoritmos Genéticos en la Optimización y Búsqueda.

Año	Investigadores	Descripción
<b>BIOLOGY</b>		
1967	Rosenberg	Simulation of the evolution of single-celled organism populations
1970	Weinberg	Outline of cell population simulation including metalevel GA
1984	Perry	Investigation of niche theory and speciation with GAs
1985	Grosso	Simulation of diploid GA with explicit subpopulation and migration
1987	Sannier y Goodman	GA adapts structures responding to spatial and temporal food availability
<b>COMPUTER SCIENCE</b>		
1967	Bagley	Parameter search in hexapawn-like game evaluation game via GA
1979	Raghavan and Birchard	GA-based clustering Algorithm
1982	Gerardy	Probabilistic automaton identification attempt via GA
1984	Gordon	Adaptive document description using GA
1985	Redndell	GA search for game evaluation function
1987	Raghavan and Agarwal	Adaptive document clustering using GA
<b>ENGINEERING AND OPERATIONS RESEARCH</b>		
1981	Goldberg	Mass-spring-dashpot system identification with simple GA
1982	Etter, Hicks and Cho	Recursive adaptive filter design using GA
1983	Goldberg	Steady-state and transit optimization of gas pipeline using GA
1985	Davis	Bin-packing and graph-coloring problems via GA
1985	Davis	Outline of job shop scheduling procedure via GA
1985	Davis and Smith	VSLI circuit layout via GA
1985	Fourman	VSLI compaction via GA
1985	Goldberg band Kuo	On-off, steady-state optimization of oil pump-pipeline system via GA
1986	Glover	Keyboard configuration design using a GA
1986	Goldberg and Samtani	Structural optimization (plane truss) via GA

**Tabla 26. Aplicaciones de los Algoritmos Genéticos  
en la Optimización y Búsqueda. (Continuación)**

1986	Goldberg and Smith	Blind knapsack problem with simple GA
1986	Minga	Aircraft landing strut weight optimization with GA
1987	Davis and Coombs	Communications link size optimization using GA plus advanced operators
	<b>GENETIC ALGORITHMS</b>	
1962	Holland	Outline for adaptive systems with programs roving cellular computer
1968	Holland	Development of theory of schemata
1971	Hollstein	2-D function optimization with mating and selection rules
1972	Bosworth, Foo, and Zeigler	GA-like operators on real genes with sophisticated mutation
1972	Frantz	Investigation of positional nonlinearity and inversion
1973	Holland	Optimal allocation of trials in GA and the two-armed bandit problem
1973	Martin	Theoretical study of GA-like probabilistic algorithms
1975	De Jong	Baseline parametric study of simple GA in five-function in test bed
1975	Holland	Publication of ANAS
1977	Mercer	GA controlled by metalevel GA
1981	Bethke	Application of walsh functions to schema average analysis
1981	Brindle	Investigation of selection and dominance in GAs
1983	Pettit and Swigger	Cursory investigation of Gas in nonstationary search problems
1983	Wetzel	Traveling salesman problem (TSP) via GA
1984	Mauldin	Study of several heuristic to maintain diversity in simple in GA
1985	Baker	Trial of ranking selection procedure on De Jong test bed
1985	Booker	Suggestion for partial match scores, sharing, and mating restriction
1985	Goldberg and lingle	TSP using partially matched crossover (PMX) and o-schema analysis
1985	Grafenstette and Fitzpatrick	Test of simple genetic algorithm with noisy functions
1985	Schaffer	Multiobjective optimization using Gas with subpopulations

**Tabla 26. Aplicaciones de los Algoritmos Genéticos  
en la Optimización y Búsqueda. (Continuación)**

1986	Golberg	Maximize marginal schema content for optimal population
1986	Goldberg	Maximize marginal schema content for optimal population size estimate
1986	Grefenstette	GA controlled by metalevel GA
1987	Baker	Reduction of stochastic errors in selection procedures
1987	Bridges and Goldberg	Extended analysis in reproduction and crossover in <i>l</i> -bit GA
1987	Goldberg	The minimal deceptive problem (MDP) under reproduction and crossover
1987	Goldberg and Richardson	Niche and species induction using sharing functions
1987	Goldberg and Segrest	Finite markov chain analysis of reproduction and mutation
1987	Goldberg and Smith	Nonstationary function optimization using diploid GAs
1987	Oliver, Smith , and Holland	Simulation and analysis permutation recombination operator
1987	Schaffer	Analysis of selection procedure effects on schema sampling
1987	Scheffer and Morishima	String-encoding adaptive crossover trial
<b>HYBRID TECHNIQUES</b>		
1985	Ackley	Connectionist algorithm with GA-like properties claimed
1985	Brady	Traveling salesman problem via genetic-like operators
1985	Grefenstette et al.	TSP via knowledge-augmented genetic operators
1987	Dolan and Dyer	Proposal to use GA to learn connectionist network topology
1987	Grefenstette	Using nonpayoff, problem-specific information in genetic search
1987	Liepins, Hilliard, palmer, and Morrow	Comparison of blind and greedy operators on combinatorial problems
1987	Shaefer	Globally modify adaptive representation technique (ARGOT)
1987	Sirag and Weisser	Simulated annealing-like control of genetic operator frequency in the TSP
1987	Suh and Van Gucht	Knowledge-based genetic operators in the TSP
<b>IMAGE PROCESING &amp; PATTERN RECOGNITION</b>		
1970	Cavicchio	Selection of detectors for binary pattern recognition

**Tabla 26. Aplicaciones de los Algoritmos Genéticos en la Optimización y Búsqueda. (Continuación)**

1984	Fitzpatrick, Grefenstette, and Van Gucht	Image registration via GA to minimize image differences
1985	Gillies	Search for image feature detectors via GA
1987	Standnyk	Explicit pattern class recognition using partial matching
<b>PARALLEL GA IMPLEMENTATIONS</b>		
1976	bethke	Brief theoretical investigation of possible parallel GA implementations
1981	Grefenstette	Brief theoretical investigation of several parallel GA implementations
1987	Cohoon, Hedge, Martin, and Richards	Simulated parallel implementation of optimal linear arrangement
1987	Jog and Van Gucht	Combined knowledge-based and parallelized GA
1987	Petty, Leuze, and Grefenstette	Parallel GA implementation on Intel hardware using De Jong test bed
1987	Suh and Van Gucht	Localized selection in parallel GA search on the TSP
1987	Tanese	Parallel GA implemented on 64-NCUBE processor
<b>PHYSICAL SCIENCES</b>		
1985	Shaefer	Nonlinear equation solving with GA for fitting potential surfaces
<b>SOCIAL SCIENCES</b>		
1979	Reynolds	GA-like adaptation in model of prehistoric hunter-gatherer behavior
1981	Smith and De Jong	Calibration of population migration model using GA search
1985	Axelrod	Simulation of the evolution of behavioral norms with GA
1985	Axelrod	Iterated prisoner's dilemma problem solution using GA

**Fuente: Genetic Algorithms in Search Optimization, and Machine Learning.**

## Anexo D

A continuación, se presenta una tabla de comparación de valores correspondientes a las probabilidades de cruce y mutación (parámetros de entrada), que de acuerdo a la revisión corta de la literatura, son expuestas por los autores como las más usadas y sugeridas en el desarrollo de experimentos para el buen funcionamiento de la metaheurística de búsqueda: *el Algoritmo Genético (AG)*.

**Tabla 27. Probabilidades de Cruce y Mutación en la Literatura.**

LITERATURA	PROBABILIDAD DE CRUCE		PROBABILIDAD DE MUTACIÓN	
	MIN	MAX	MIN	MAX
AHMED, 2010. [1]	--	1,00	--	0,01
AHMED, 2013. [2]	--	1,00	--	0,02
BENREJEB, 2011. [8]	--	0,70	--	0,01
BHANDARI 2012. [9]	0,75	1,00	0,001	0,01
DAIDA, 1998. [15]	--	0,60	--	0,03
FIDANOVA, 2013. [24]	--	1,00	--	0,01
JINGUI, 2010. [30]	0,80	0,90	0,05	0,20
PAPUTUNGAN, 2013. [40]	--	0,50	--	0,05
CHUNGUO, 2008. [12]	--	0,89	--	0,03
DONG, 2009. [23]	--	0,8	--	0,1
AVILA-DÍAZ, 2008. [5]	--	0,6	--	0,01

Fuente: Consulta de Literatura.

Anexo E

Tabla 28. Parámetros de Entrada del Algoritmo Genético Propuesto (AG-sTSP).

<b><i>Parámetro de Entrada del AG</i></b>	<b><i>Descripción</i></b>
<b><i>PosX</i></b>	Identifica la ubicación de la fila del dato (0,0) de la instancia convertida.
<b><i>PosY</i></b>	Identifica la ubicación de la columna del dato (0,0) de la instancia convertida.
<b><i>P_Distrib</i></b>	Número de sitios que debe visitar el agente viajero sin tener en cuenta el deposito o lugar de partida.
<b><i>Tam_Pob</i></b>	Número de individuos que conforman la población de tratamiento para el procesamiento de la instancia.
<b><i>Prop_Pob_Cruce</i></b>	Proporción de individuos que aportan los procesos de recombinación para la conformación de la nueva población de tratamiento (60% del Tamaño de la Población Inicial).
<b><i>Prop_Pob_Mutacion</i></b>	Proporción de individuos que aporta el proceso de alteración para la conformación de una parte de la nueva población de tratamiento (20% del Tamaño de la Población Inicial).
<b><i>Prop_Pob_Elite</i></b>	Proporción de individuos que conforman la población élite (2% del Tamaño de la Población Inicial) con un valor mínimo de 2 individuos.
<b><i>Prob_Cruce</i></b>	Probabilidad de ocurrencia del proceso de cruce probabilístico aplicado a 2 individuos de la población de tratamiento.
<b><i>Prob_Mutacion</i></b>	Probabilidad de ocurrencia del proceso de mutación SIM aplicado a 1 individuo de la población de tratamiento.
<b><i>Prob_Elite</i></b>	Probabilidad de ocurrencia de la inserción del individuo élite en la nueva población de tratamiento.
<b><i>Tenencia_Elite</i></b>	Criterio que libera el mejor individuo élite para su posterior remplazo por aquel que le individuo que le precede.

Fuente: Elaborado por el Autor.

Anexo F

Tabla 29. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia bays29.

Configuración 1:	<i>Tpob</i>	<i>Pc</i>	<i>Pm</i>	<i>IAG</i>	<i>Te</i>
Nivel Inferior	29	0,75	0,001	170000	--
Nivel Superior	43	1	0,03	200000	150

Configuración 2:	<i>Tpob</i>	<i>Pc</i>	<i>Pm</i>	<i>IAG</i>	<i>Te</i>
Nivel Inferior	29	0,75	0,001	170000	--
Nivel Superior	43	1	0,03	200000	300

Configuración 3:	<i>Tpob</i>	<i>Pc</i>	<i>Pm</i>	<i>IAG</i>	<i>Te</i>
Nivel Inferior	29	0,75	0,06	170000	--
Nivel Superior	43	1	0,1	200000	150

Configuración 4:	<i>Tpob</i>	<i>Pc</i>	<i>Pm</i>	<i>IAG</i>	<i>Te</i>
Nivel Inferior	29	0,75	0,06	170000	--
Nivel Superior	43	1	0,1	200000	300

Fuente: Elaborado por el Autor.

Tabla 30. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia bays29.

PARÁMETROS	AG-sTSP	# Rep.	Prop AG-sTSP	Z Opt Gap	Ts-TSP Gap	CnW Gap
Configuración 1	2026	7	10,94%	-0,30%	0,00%	24,52%
Configuración 2	2026	9	14,06%	-0,30%	0,00%	24,52%
	2022	4	6,25%	-0,10%	0,20%	24,66%
Configuración 3	2020	2	3,13%	0,00%	0,30%	24,74%
	2026	11	17,19%	-0,30%	0,00%	24,52%
	2022	2	3,13%	-0,10%	0,20%	24,66%
Configuración 4	2020	1	1,56%	0,00%	0,30%	24,74%
	2026	18	28,13%	-0,30%	0,00%	24,52%
	2022	1	1,56%	-0,10%	0,20%	24,66%
	2020	5	7,81%	0,00%	0,30%	24,74%

Fuente: Elaborado por el Autor.

Tabla 31. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia ei51.

Configuración 1:	<i>T<sub>pob</sub></i>	<i>P<sub>c</sub></i>	<i>P<sub>m</sub></i>	<i>IAG</i>	<i>T<sub>e</sub></i>
Nivel Inferior	51	0,75	0,06	150000	--
Nivel Superior	76	1	0,1	180000	300

Configuración 2:	<i>T<sub>pob</sub></i>	<i>P<sub>c</sub></i>	<i>P<sub>m</sub></i>	<i>IAG</i>	<i>T<sub>e</sub></i>
Nivel Inferior	51	0,75	0,06	150000	--
Nivel Superior	76	1	0,1	180000	370

Configuración 3:	<i>T<sub>pob</sub></i>	<i>P<sub>c</sub></i>	<i>P<sub>m</sub></i>	<i>IAG</i>	<i>T<sub>e</sub></i>
Nivel Inferior	51	0,75	0,06	250000	--
Nivel Superior	76	1	0,1	300000	300

Configuración 4:	<i>T<sub>pob</sub></i>	<i>P<sub>c</sub></i>	<i>P<sub>m</sub></i>	<i>IAG</i>	<i>T<sub>e</sub></i>
Nivel Inferior	51	0,75	0,06	250000	--
Nivel Superior	76	1	0,1	300000	370

Fuente: Elaborado por el Autor.

Tabla 32. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia ei51.

PARÁMETROS	AG-sTSP	# Rep.	Prop AG-sTSP	Z Opt Gap	Ts-TSP Gap	CnW Gap
Configuración 1	455	1	1,56%	-6,81%	0,00%	16,36%
	454	4	6,25%	-6,57%	0,22%	16,54%
	453	5	7,81%	-6,34%	0,44%	16,73%
	450	1	1,56%	-5,63%	1,10%	17,28%
	449	3	4,69%	-5,40%	1,32%	17,46%
	446	2	3,13%	-4,69%	1,98%	18,01%
	443	1	1,56%	-3,99%	2,64%	18,57%
Configuración 2	442	1	1,56%	-3,76%	2,86%	18,75%
	455	3	4,69%	-6,81%	0,00%	16,36%
	453	2	3,13%	-6,34%	0,44%	16,73%
	452	1	1,56%	-6,10%	0,66%	16,91%
	451	2	3,13%	-5,87%	0,88%	17,10%
	449	1	1,56%	-5,40%	1,32%	17,46%
Configuración 3	448	4	6,25%	-5,16%	1,54%	17,65%
	455	5	7,81%	-6,81%	0,00%	16,36%
	453	2	3,13%	-6,34%	0,44%	16,73%
	452	7	10,94%	-6,10%	0,66%	16,91%
	451	3	4,69%	-5,87%	0,88%	17,10%
	450	1	1,56%	-5,63%	1,10%	17,28%
449	3	4,69%	-5,40%	1,32%	17,46%	

**Tabla 32. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia eil51. (Continuación).**

<b>Configuración 3</b>	447	2	3,13%	-4,93%	1,76%	17,83%
	446	3	4,69%	-4,69%	1,98%	18,01%
	445	1	1,56%	-4,46%	2,20%	18,20%
	443	3	4,69%	-3,99%	2,64%	18,57%
	441	1	1,56%	-3,52%	3,08%	18,93%
<b>Configuración 4</b>	455	4	6,25%	-6,81%	0,00%	16,36%
	454	3	4,69%	-6,57%	0,22%	16,54%
	453	3	4,69%	-6,34%	0,44%	16,73%
	452	1	1,56%	-6,10%	0,66%	16,91%
	451	4	6,25%	-5,87%	0,88%	17,10%
	450	1	1,56%	-5,63%	1,10%	17,28%
	449	1	1,56%	-5,40%	1,32%	17,46%
	448	2	3,13%	-5,16%	1,54%	17,65%
	447	1	1,56%	-4,93%	1,76%	17,83%
	446	2	3,13%	-4,69%	1,98%	18,01%
	445	1	1,56%	-4,46%	2,20%	18,20%
	444	3	4,69%	-4,23%	2,42%	18,38%
	443	2	3,13%	-3,99%	2,64%	18,57%
	442	2	3,13%	-3,76%	2,86%	18,75%
	441	1	1,56%	-3,52%	3,08%	18,93%
440	1	1,56%	-3,29%	3,30%	19,12%	

Fuente: Elaborado por el Autor.

**Tabla 33. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia st70.**

<b>Configuración 1:</b>	<b><i>T<sub>pob</sub></i></b>	<b><i>P<sub>c</sub></i></b>	<b><i>P<sub>m</sub></i></b>	<b><i>IAG</i></b>	<b><i>T<sub>e</sub></i></b>
<b>Nivel Inferior</b>	70	0,75	0,001	350000	--
<b>Nivel Superior</b>	105	0,1	0,03	500000	250

<b>Configuración 2:</b>	<b><i>T<sub>pob</sub></i></b>	<b><i>P<sub>c</sub></i></b>	<b><i>P<sub>m</sub></i></b>	<b><i>IAG</i></b>	<b><i>T<sub>e</sub></i></b>
<b>Nivel Inferior</b>	70	0,75	0,001	350000	--
<b>Nivel Superior</b>	105	0,1	0,03	500000	500

<b>Configuración 3:</b>	<b><i>T<sub>pob</sub></i></b>	<b><i>P<sub>c</sub></i></b>	<b><i>P<sub>m</sub></i></b>	<b><i>IAG</i></b>	<b><i>T<sub>e</sub></i></b>
<b>Nivel Inferior</b>	70	0,75	0,06	350000	--
<b>Nivel Superior</b>	105	0,1	0,1	500000	250

<b>Configuración 4:</b>	<b><i>T<sub>pob</sub></i></b>	<b><i>P<sub>c</sub></i></b>	<b><i>P<sub>m</sub></i></b>	<b><i>IAG</i></b>	<b><i>T<sub>e</sub></i></b>
<b>Nivel Inferior</b>	70	0,75	0,06	350000	--
<b>Nivel Superior</b>	105	0,1	0,1	500000	500

Fuente: Elaborado por el Autor.

**Tabla 34. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia st70.**

PARÁMETROS	AG-sTSP	# Rep.	Prop AG-sTSP	Z Opt Gap	Ts-TSP Gap	CnW Gap
Configuración 1	738	3	4,69%	-9,33%	0,00%	22,23%
	736	2	3,13%	-9,04%	0,27%	22,44%
	734	2	3,13%	-8,74%	0,54%	22,66%
	733	1	1,56%	-8,59%	0,68%	22,76%
	731	1	1,56%	-8,30%	0,95%	22,97%
	730	1	1,56%	-8,15%	1,08%	23,08%
	726	1	1,56%	-7,56%	1,63%	23,50%
	725	3	4,69%	-7,41%	1,76%	23,60%
	723	1	1,56%	-7,11%	2,03%	23,81%
	722	1	1,56%	-6,96%	2,17%	23,92%
	718	3	4,69%	-6,37%	2,71%	24,34%
	718	3	4,69%	-6,37%	2,71%	24,34%
	716	1	1,56%	-6,07%	2,98%	24,55%
	715	1	1,56%	-5,93%	3,12%	24,66%
	714	1	1,56%	-5,78%	3,25%	24,76%
711	1	1,56%	-5,33%	3,66%	25,08%	
Configuración 2	737	1	1,56%	-9,19%	0,14%	22,34%
	735	3	4,69%	-8,89%	0,41%	22,55%
	734	1	1,56%	-8,74%	0,54%	22,66%
	730	4	6,25%	-8,15%	1,08%	23,08%
	729	1	1,56%	-8,00%	1,22%	23,18%
	728	2	3,13%	-7,85%	1,36%	23,29%
	727	1	1,56%	-7,70%	1,49%	23,39%
	726	3	4,69%	-7,56%	1,63%	23,50%
	724	1	1,56%	-7,26%	1,90%	23,71%
	723	3	4,69%	-7,11%	2,03%	23,81%
	721	1	1,56%	-6,81%	2,30%	24,03%
	718	1	1,56%	-6,37%	2,71%	24,34%
	717	2	3,13%	-6,22%	2,85%	24,45%
	716	1	1,56%	-6,07%	2,98%	24,55%
	715	5	7,81%	-5,93%	3,12%	24,66%
	713	2	3,13%	-5,63%	3,39%	24,87%
	712	3	4,69%	-5,48%	3,52%	24,97%
	710	3	4,69%	-5,19%	3,79%	25,18%
708	1	1,56%	-4,89%	4,07%	25,40%	
706	2	3,13%	-4,59%	4,34%	25,61%	

**Tabla 34. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia st70. (Continuación).**

<b>Configuración 2</b>	705	1	1,56%	-4,44%	4,47%	25,71%
	704	1	1,56%	-4,30%	4,61%	25,82%
	703	1	1,56%	-4,15%	4,74%	25,92%
	702	1	1,56%	-4,00%	4,88%	26,03%
	701	1	1,56%	-3,85%	5,01%	26,13%
	700	1	1,56%	-3,70%	5,15%	26,24%
	697	1	1,56%	-3,26%	5,56%	26,55%
<b>Configuración 3</b>	738	1	1,56%	-9,33%	0,00%	22,23%
	737	3	4,69%	-9,19%	0,14%	22,34%
	736	1	1,56%	-9,04%	0,27%	22,44%
	735	2	3,13%	-8,89%	0,41%	22,55%
	733	2	3,13%	-8,59%	0,68%	22,76%
	728	3	4,69%	-7,85%	1,36%	23,29%
	727	2	3,13%	-7,70%	1,49%	23,39%
	726	2	3,13%	-7,56%	1,63%	23,50%
	723	1	1,56%	-7,11%	2,03%	23,81%
	721	1	1,56%	-6,81%	2,30%	24,03%
	720	3	4,69%	-6,67%	2,44%	24,13%
	715	2	3,13%	-5,93%	3,12%	24,66%
	712	1	1,56%	-5,48%	3,52%	24,97%
	710	1	1,56%	-5,19%	3,79%	25,18%
<b>Configuración 4</b>	737	1	1,56%	-9,19%	0,14%	22,34%
	736	2	3,13%	-9,04%	0,27%	22,44%
	734	2	3,13%	-8,74%	0,54%	22,66%
	732	3	4,69%	-8,44%	0,81%	22,87%
	729	1	1,56%	-8,00%	1,22%	23,18%
	728	7	10,94%	-7,85%	1,36%	23,29%
	727	1	1,56%	-7,70%	1,49%	23,39%
	725	3	4,69%	-7,41%	1,76%	23,60%
	724	2	3,13%	-7,26%	1,90%	23,71%
	722	2	3,13%	-6,96%	2,17%	23,92%
	721	2	3,13%	-6,81%	2,30%	24,03%
	720	2	3,13%	-6,67%	2,44%	24,13%
	719	1	1,56%	-6,52%	2,57%	24,24%
	718	2	3,13%	-6,37%	2,71%	24,34%
	716	3	4,69%	-6,07%	2,98%	24,55%
	715	1	1,56%	-5,93%	3,12%	24,66%

**Tabla 35. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia st70. (Continuación).**

<b>Configuración 4</b>	714	1	1,56%	-5,78%	3,25%	24,76%
	713	2	3,13%	-5,63%	3,39%	24,87%
	712	1	1,56%	-5,48%	3,52%	24,97%
	711	1	1,56%	-5,33%	3,66%	25,08%
	708	2	3,13%	-4,89%	4,07%	25,40%
	707	2	3,13%	-4,74%	4,20%	25,50%
	705	1	1,56%	-4,44%	4,47%	25,71%
	704	1	1,56%	-4,30%	4,61%	25,82%
	702	2	3,13%	-4,00%	4,88%	26,03%
	700	2	3,13%	-3,70%	5,15%	26,24%
	698	1	1,56%	-3,41%	5,42%	26,45%

Fuente: Elaborado por el Autor.

**Tabla 35. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia eil76.**

<b>Configuración 1:</b>	<b><i>Tpob</i></b>	<b><i>Pc</i></b>	<b><i>Pm</i></b>	<b><i>IAG</i></b>	<b><i>Te</i></b>
<b>Nivel Inferior</b>	76	0,75	0,001	350000	--
<b>Nivel Superior</b>	114	1	0,03	500000	250

<b>Configuración 2:</b>	<b><i>Tpob</i></b>	<b><i>Pc</i></b>	<b><i>Pm</i></b>	<b><i>IAG</i></b>	<b><i>Te</i></b>
<b>Nivel Inferior</b>	76	0,75	0,001	350000	--
<b>Nivel Superior</b>	114	1	0,03	500000	500

<b>Configuración 3:</b>	<b><i>Tpob</i></b>	<b><i>Pc</i></b>	<b><i>Pm</i></b>	<b><i>IAG</i></b>	<b><i>Te</i></b>
<b>Nivel Inferior</b>	76	0,75	0,06	350000	--
<b>Nivel Superior</b>	114	1	0,01	500000	250

<b>Configuración 4:</b>	<b><i>Tpob</i></b>	<b><i>Pc</i></b>	<b><i>Pm</i></b>	<b><i>IAG</i></b>	<b><i>Te</i></b>
<b>Nivel Inferior</b>	76	0,75	0,06	350000	--
<b>Nivel Superior</b>	114	1	0,01	500000	500

Fuente: Elaborado por el Autor.

**Tabla 36. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia eil76.**

<b>PARÁMETROS</b>	<b>AG-sTSP</b>	<b># Rep.</b>	<b>Prop AG-sTSP</b>	<b>Z Opt Gap</b>	<b>Ts-TSP Gap</b>	<b>CnW Gap</b>
<b>Configuración 1</b>	578	2	3,13%	-7,43%	0,34%	16,59%
	577	3	4,69%	-7,25%	0,52%	16,74%
	576	5	7,81%	-7,06%	0,69%	16,88%
	575	3	4,69%	-6,88%	0,86%	17,03%

**Tabla 36. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia eil76. (Continuación).**

<b>Configuración 1</b>	573	1	1,56%	-6,51%	1,21%	17,32%
	572	1	1,56%	-6,32%	1,38%	17,46%
	571	2	3,13%	-6,13%	1,55%	17,60%
	570	1	1,56%	-5,95%	1,72%	17,75%
	569	1	1,56%	-5,76%	1,90%	17,89%
	567	1	1,56%	-5,39%	2,24%	18,18%
	565	1	1,56%	-5,02%	2,59%	18,47%
	558	1	1,56%	-3,72%	3,79%	19,48%
<b>Configuración 2</b>	580	1	1,56%	-7,81%	0,00%	16,31%
	579	1	1,56%	-7,62%	0,17%	16,45%
	578	1	1,56%	-7,43%	0,34%	16,59%
	577	3	4,69%	-7,25%	0,52%	16,74%
	575	3	4,69%	-6,88%	0,86%	17,03%
	574	1	1,56%	-6,69%	1,03%	17,17%
	573	1	1,56%	-6,51%	1,21%	17,32%
	572	3	4,69%	-6,32%	1,38%	17,46%
	571	3	4,69%	-6,13%	1,55%	17,60%
	570	1	1,56%	-5,95%	1,72%	17,75%
	569	5	7,81%	-5,76%	1,90%	17,89%
	567	2	3,13%	-5,39%	2,24%	18,18%
	566	1	1,56%	-5,20%	2,41%	18,33%
	565	2	3,13%	-5,02%	2,59%	18,47%
	563	3	4,69%	-4,65%	2,93%	18,76%
	561	1	1,56%	-4,28%	3,28%	19,05%
	559	2	3,13%	-3,90%	3,62%	19,34%
	557	1	1,56%	-3,53%	3,97%	19,62%
554	1	1,56%	-2,97%	4,48%	20,06%	
<b>Configuración 3</b>	580	2	3,13%	-7,81%	0,00%	16,31%
	579	2	3,13%	-7,62%	0,17%	16,45%
	578	2	3,13%	-7,43%	0,34%	16,59%
	577	3	4,69%	-7,25%	0,52%	16,74%
	576	3	4,69%	-7,06%	0,69%	16,88%
	575	3	4,69%	-6,88%	0,86%	17,03%
	574	1	1,56%	-6,69%	1,03%	17,17%
	572	1	1,56%	-6,32%	1,38%	17,46%
	571	3	4,69%	-6,13%	1,55%	17,60%
	564	1	1,56%	-4,83%	2,76%	18,61%

**Tabla 36. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia eil76. (Continuación).**

<b>Configuración 4</b>	562	1	1,56%	-4,46%	3,10%	18,90%
	555	1	1,56%	-3,16%	4,31%	19,91%
	580	2	3,13%	-7,81%	0,00%	16,31%
	579	3	4,69%	-7,62%	0,17%	16,45%
	578	1	1,56%	-7,43%	0,34%	16,59%
	577	3	4,69%	-7,25%	0,52%	16,74%
	576	2	3,13%	-7,06%	0,69%	16,88%
	575	3	4,69%	-6,88%	0,86%	17,03%
	574	1	1,56%	-6,69%	1,03%	17,17%
	573	2	3,13%	-6,51%	1,21%	17,32%
	572	4	6,25%	-6,32%	1,38%	17,46%
	571	1	1,56%	-6,13%	1,55%	17,60%
	570	5	7,81%	-5,95%	1,72%	17,75%
	568	2	3,13%	-5,58%	2,07%	18,04%
	567	1	1,56%	-5,39%	2,24%	18,18%
	566	2	3,13%	-5,20%	2,41%	18,33%
	565	1	1,56%	-5,02%	2,59%	18,47%
	564	1	1,56%	-4,83%	2,76%	18,61%
	563	1	1,56%	-4,65%	2,93%	18,76%
	562	1	1,56%	-4,46%	3,10%	18,90%
	561	1	1,56%	-4,28%	3,28%	19,05%
	560	1	1,56%	-4,09%	3,45%	19,19%
	559	1	1,56%	-3,90%	3,62%	19,34%
	546	1	1,56%	-1,49%	5,86%	21,21%

Fuente: Elaborado por el Autor.

**Tabla 37. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia pr76.**

<b>Configuración 1:</b>	<b><i>T<sub>pob</sub></i></b>	<b><i>P<sub>c</sub></i></b>	<b><i>P<sub>m</sub></i></b>	<b><i>IAG</i></b>	<b><i>T<sub>e</sub></i></b>
<b>Nivel Inferior</b>	76	0,75	0,001	350000	--
<b>Nivel Superior</b>	114	1	0,03	500000	250

<b>Configuración 2:</b>	<b><i>T<sub>pob</sub></i></b>	<b><i>P<sub>c</sub></i></b>	<b><i>P<sub>m</sub></i></b>	<b><i>IAG</i></b>	<b><i>T<sub>e</sub></i></b>
<b>Nivel Inferior</b>	76	0,75	0,001	350000	--
<b>Nivel Superior</b>	114	1	0,03	500000	500

<b>Configuración 3:</b>	<b><i>T<sub>pob</sub></i></b>	<b><i>P<sub>c</sub></i></b>	<b><i>P<sub>m</sub></i></b>	<b><i>IAG</i></b>	<b><i>T<sub>e</sub></i></b>
<b>Nivel Inferior</b>	76	0,75	0,06	350000	--
<b>Nivel Superior</b>	114	1	0,01	500000	250

**Tabla 37. Grupo de Configuraciones Tentativas de los Parámetros de Entrada del AG Propuesto, para la Instancia pr76. (Continuación)**

Configuración 4:	<i>T<sub>pob</sub></i>	<i>P<sub>c</sub></i>	<i>P<sub>m</sub></i>	<i>IAG</i>	<i>T<sub>e</sub></i>
Nivel Inferior	76	0,75	0,06	350000	--
Nivel Superior	114	1	0,01	500000	500

Fuente: Elaborado por el Autor.

**Tabla 38. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia pr76.**

PARÁMETROS	AG-sTSP	# Rep.	Prop AG-sTSP	Z Opt Gap	Ts-TSP Gap	CnW
Configuración 1	124670	1	1,56%	-15,27%	0,90%	17,74%
	124545	1	1,56%	-15,15%	1,00%	17,82%
	124070	1	1,56%	-14,71%	1,37%	18,14%
	123786	1	1,56%	-14,45%	1,60%	18,32%
	123716	1	1,56%	-14,38%	1,66%	18,37%
	123696	1	1,56%	-14,36%	1,67%	18,38%
	123309	1	1,56%	-14,01%	1,98%	18,64%
	122683	1	1,56%	-13,43%	2,48%	19,05%
	122287	1	1,56%	-13,06%	2,79%	19,31%
	122121	1	1,56%	-12,91%	2,92%	19,42%
	122002	1	1,56%	-12,80%	3,02%	19,50%
	121982	1	1,56%	-12,78%	3,03%	19,51%
	121895	1	1,56%	-12,70%	3,10%	19,57%
	121578	1	1,56%	-12,41%	3,36%	19,78%
	121461	1	1,56%	-12,30%	3,45%	19,86%
	121313	1	1,56%	-12,16%	3,57%	19,95%
	121049	1	1,56%	-11,92%	3,78%	20,13%
	120740	3	4,69%	-11,63%	4,02%	20,33%
	120393	2	3,13%	-11,31%	4,30%	20,56%
	119926	1	1,56%	-10,88%	4,67%	20,87%
	119786	1	1,56%	-10,75%	4,78%	20,96%
	119678	1	1,56%	-10,65%	4,87%	21,03%
	119616	1	1,56%	-10,59%	4,91%	21,07%
	119557	1	1,56%	-10,54%	4,96%	21,11%
	119533	1	1,56%	-10,52%	4,98%	21,13%
	119151	1	1,56%	-10,16%	5,28%	21,38%
	118910	1	1,56%	-9,94%	5,48%	21,54%
	118813	1	1,56%	-9,85%	5,55%	21,60%
	118738	1	1,56%	-9,78%	5,61%	21,65%

**Tabla 38. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia pr76. (Continuación).**

<b>Configuración 1</b>	118563	1	1,56%	-9,62%	5,75%	21,77%
	118327	1	1,56%	-9,40%	5,94%	21,92%
	117904	1	1,56%	-9,01%	6,28%	22,20%
	117681	1	1,56%	-8,80%	6,45%	22,35%
	117586	1	1,56%	-8,72%	6,53%	22,41%
	117333	1	1,56%	-8,48%	6,73%	22,58%
	117302	2	3,13%	-8,45%	6,75%	22,60%
	117192	1	1,56%	-8,35%	6,84%	22,67%
	116852	1	1,56%	-8,04%	7,11%	22,90%
	116482	1	1,56%	-7,70%	7,41%	23,14%
	115891	1	1,56%	-7,15%	7,88%	23,53%
	115162	1	1,56%	-6,47%	8,46%	24,01%
	114988	1	1,56%	-6,31%	8,59%	24,13%
	114749	1	1,56%	-6,09%	8,78%	24,29%
	114563	1	1,56%	-5,92%	8,93%	24,41%
	113757	1	1,56%	-5,18%	9,57%	24,94%
	113641	1	1,56%	-5,07%	9,66%	25,02%
	112715	1	1,56%	-4,21%	10,40%	25,63%
<b>Configuración 2</b>	125075	1	1,56%	-15,64%	0,58%	17,47%
	124591	1	1,56%	-15,19%	0,96%	17,79%
	123714	1	1,56%	-14,38%	1,66%	18,37%
	123567	1	1,56%	-14,25%	1,77%	18,47%
	123038	1	1,56%	-13,76%	2,19%	18,82%
	122253	1	1,56%	-13,03%	2,82%	19,33%
	121528	1	1,56%	-12,36%	3,40%	19,81%
	121416	1	1,56%	-12,26%	3,48%	19,89%
	121041	1	1,56%	-11,91%	3,78%	20,13%
	120515	1	1,56%	-11,42%	4,20%	20,48%
	120166	1	1,56%	-11,10%	4,48%	20,71%
	119946	1	1,56%	-10,90%	4,65%	20,86%
	119916	1	1,56%	-10,87%	4,68%	20,88%
	119729	1	1,56%	-10,70%	4,83%	21,00%
	119650	1	1,56%	-10,62%	4,89%	21,05%
	119309	1	1,56%	-10,31%	5,16%	21,28%
	119229	1	1,56%	-10,23%	5,22%	21,33%
	119196	1	1,56%	-10,20%	5,25%	21,35%
119115	1	1,56%	-10,13%	5,31%	21,40%	

**Tabla 38. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia pr76. (Continuación).**

<b>Configuración 2</b>	118929	1	1,56%	-9,96%	5,46%	21,53%
	118820	1	1,56%	-9,86%	5,55%	21,60%
	118595	1	1,56%	-9,65%	5,73%	21,75%
	118331	1	1,56%	-9,40%	5,94%	21,92%
	118127	1	1,56%	-9,22%	6,10%	22,06%
	118121	1	1,56%	-9,21%	6,10%	22,06%
	117942	1	1,56%	-9,05%	6,25%	22,18%
	117848	1	1,56%	-8,96%	6,32%	22,24%
	117687	1	1,56%	-8,81%	6,45%	22,35%
	117533	1	1,56%	-8,67%	6,57%	22,45%
	117366	1	1,56%	-8,51%	6,70%	22,56%
	117301	1	1,56%	-8,45%	6,76%	22,60%
	117234	1	1,56%	-8,39%	6,81%	22,65%
	117214	1	1,56%	-8,37%	6,82%	22,66%
	117106	1	1,56%	-8,27%	6,91%	22,73%
	117007	1	1,56%	-8,18%	6,99%	22,80%
	116716	1	1,56%	-7,91%	7,22%	22,99%
	116614	1	1,56%	-7,82%	7,30%	23,05%
	116258	1	1,56%	-7,49%	7,58%	23,29%
	116207	1	1,56%	-7,44%	7,62%	23,32%
	116175	1	1,56%	-7,41%	7,65%	23,34%
	116122	1	1,56%	-7,36%	7,69%	23,38%
	115933	1	1,56%	-7,19%	7,84%	23,50%
	115914	1	1,56%	-7,17%	7,86%	23,52%
	115793	1	1,56%	-7,06%	7,95%	23,60%
	115686	1	1,56%	-6,96%	8,04%	23,67%
	115619	1	1,56%	-6,90%	8,09%	23,71%
	115315	1	1,56%	-6,62%	8,33%	23,91%
	115304	1	1,56%	-6,61%	8,34%	23,92%
	115276	1	1,56%	-6,58%	8,36%	23,94%
	115254	1	1,56%	-6,56%	8,38%	23,95%
	114950	1	1,56%	-6,28%	8,62%	24,15%
114354	1	1,56%	-5,73%	9,10%	24,55%	
114004	1	1,56%	-5,40%	9,38%	24,78%	
113831	1	1,56%	-5,24%	9,51%	24,89%	
113774	1	1,56%	-5,19%	9,56%	24,93%	
113616	1	1,56%	-5,05%	9,68%	25,03%	

**Tabla 38. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia pr76. (Continuación).**

<b>Configuración 3</b>	124648	1	1,56%	-15,25%	0,91%	17,75%
	124162	1	1,56%	-14,80%	1,30%	18,07%
	123950	1	1,56%	-14,60%	1,47%	18,21%
	123890	1	1,56%	-14,54%	1,52%	18,25%
	123589	1	1,56%	-14,27%	1,76%	18,45%
	122963	1	1,56%	-13,69%	2,25%	18,87%
	122758	1	1,56%	-13,50%	2,42%	19,00%
	122732	1	1,56%	-13,47%	2,44%	19,02%
	122685	1	1,56%	-13,43%	2,48%	19,05%
	122666	1	1,56%	-13,41%	2,49%	19,06%
	122608	1	1,56%	-13,36%	2,54%	19,10%
	122511	1	1,56%	-13,27%	2,61%	19,16%
	121604	1	1,56%	-12,43%	3,33%	19,76%
	121312	1	1,56%	-12,16%	3,57%	19,96%
	121091	1	1,56%	-11,96%	3,74%	20,10%
	120823	1	1,56%	-11,71%	3,96%	20,28%
	120578	1	1,56%	-11,48%	4,15%	20,44%
	120421	1	1,56%	-11,34%	4,28%	20,54%
	120274	1	1,56%	-11,20%	4,39%	20,64%
	119784	1	1,56%	-10,75%	4,78%	20,96%
	119509	1	1,56%	-10,49%	5,00%	21,14%
	119245	1	1,56%	-10,25%	5,21%	21,32%
	119160	1	1,56%	-10,17%	5,28%	21,38%
	118876	1	1,56%	-9,91%	5,50%	21,56%
	118771	1	1,56%	-9,81%	5,59%	21,63%
	118509	1	1,56%	-9,57%	5,79%	21,80%
	118417	1	1,56%	-9,48%	5,87%	21,87%
	118396	1	1,56%	-9,46%	5,88%	21,88%
	118342	1	1,56%	-9,41%	5,93%	21,91%
	118222	1	1,56%	-9,30%	6,02%	21,99%
	118116	1	1,56%	-9,21%	6,11%	22,06%
	118049	1	1,56%	-9,14%	6,16%	22,11%
	118019	1	1,56%	-9,12%	6,18%	22,13%
	117969	1	1,56%	-9,07%	6,22%	22,16%
117930	1	1,56%	-9,03%	6,26%	22,19%	
117703	1	1,56%	-8,82%	6,44%	22,34%	
117501	1	1,56%	-8,64%	6,60%	22,47%	

**Tabla 38. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia pr76. (Continuación).**

<b>Configuración 3</b>	117417	1	1,56%	-8,56%	6,66%	22,53%
	117123	1	1,56%	-8,29%	6,90%	22,72%
	117004	1	1,56%	-8,18%	6,99%	22,80%
	116726	1	1,56%	-7,92%	7,21%	22,98%
	116299	1	1,56%	-7,53%	7,55%	23,26%
	116270	1	1,56%	-7,50%	7,57%	23,28%
	115943	1	1,56%	-7,20%	7,83%	23,50%
	115759	1	1,56%	-7,03%	7,98%	23,62%
	115585	1	1,56%	-6,87%	8,12%	23,73%
	114862	1	1,56%	-6,20%	8,69%	24,21%
	114857	1	1,56%	-6,19%	8,70%	24,21%
	113920	1	1,56%	-5,33%	9,44%	24,83%
	113167	1	1,56%	-4,63%	10,04%	25,33%
<b>Configuración 4</b>	125110	1	1,56%	-15,67%	0,55%	17,45%
	124834	1	1,56%	-15,42%	0,77%	17,63%
	124726	1	1,56%	-15,32%	0,85%	17,70%
	124391	1	1,56%	-15,01%	1,12%	17,92%
	124033	1	1,56%	-14,68%	1,40%	18,16%
	123934	1	1,56%	-14,59%	1,48%	18,23%
	123571	1	1,56%	-14,25%	1,77%	18,46%
	123341	1	1,56%	-14,04%	1,95%	18,62%
	123246	1	1,56%	-13,95%	2,03%	18,68%
	122666	1	1,56%	-13,41%	2,49%	19,06%
	122395	1	1,56%	-13,16%	2,71%	19,24%
	121245	1	1,56%	-12,10%	3,62%	20,00%
	121090	1	1,56%	-11,96%	3,74%	20,10%
	120990	1	1,56%	-11,86%	3,82%	20,17%
	120947	1	1,56%	-11,82%	3,86%	20,20%
	120589	1	1,56%	-11,49%	4,14%	20,43%
	120585	1	1,56%	-11,49%	4,14%	20,43%
	120526	1	1,56%	-11,43%	4,19%	20,47%
	120064	1	1,56%	-11,01%	4,56%	20,78%
	119697	1	1,56%	-10,67%	4,85%	21,02%
	119494	1	1,56%	-10,48%	5,01%	21,15%
	118824	1	1,56%	-9,86%	5,54%	21,60%
	118797	1	1,56%	-9,84%	5,57%	21,61%
	118189	1	1,56%	-9,27%	6,05%	22,02%

**Tabla 38. Mejores Resultados Factibles Obtenidos por el Procesamiento AG-sTSP de la Instancia pr76. (Continuación).**

Configuración 4	118067	1	1,56%	-9,16%	6,15%	22,10%
	117971	1	1,56%	-9,07%	6,22%	22,16%
	117641	1	1,56%	-8,77%	6,48%	22,38%
	117513	1	1,56%	-8,65%	6,59%	22,46%
	117500	1	1,56%	-8,64%	6,60%	22,47%
	117492	1	1,56%	-8,63%	6,60%	22,48%
	117331	1	1,56%	-8,48%	6,73%	22,58%
	117064	1	1,56%	-8,23%	6,94%	22,76%
	116854	1	1,56%	-8,04%	7,11%	22,90%
	116761	1	1,56%	-7,95%	7,18%	22,96%
	116750	1	1,56%	-7,94%	7,19%	22,97%
	116679	1	1,56%	-7,88%	7,25%	23,01%
	116635	1	1,56%	-7,84%	7,28%	23,04%
	116276	1	1,56%	-7,50%	7,57%	23,28%
	115854	1	1,56%	-7,11%	7,91%	23,56%
	115681	1	1,56%	-6,95%	8,04%	23,67%
	115586	1	1,56%	-6,87%	8,12%	23,73%
	115562	1	1,56%	-6,84%	8,14%	23,75%
	115472	1	1,56%	-6,76%	8,21%	23,81%
	115315	1	1,56%	-6,62%	8,33%	23,91%
	114924	1	1,56%	-6,25%	8,64%	24,17%
	114858	1	1,56%	-6,19%	8,70%	24,21%
	114853	1	1,56%	-6,19%	8,70%	24,22%
	114728	1	1,56%	-6,07%	8,80%	24,30%
	114694	1	1,56%	-6,04%	8,83%	24,32%
	114404	1	1,56%	-5,77%	9,06%	24,51%
	114193	1	1,56%	-5,58%	9,23%	24,65%
	114059	1	1,56%	-5,45%	9,33%	24,74%
	113933	1	1,56%	-5,34%	9,43%	24,82%
	113768	1	1,56%	-5,19%	9,56%	24,93%
	113443	1	1,56%	-4,89%	9,82%	25,15%
	113102	1	1,56%	-4,57%	10,09%	25,37%
113009	1	1,56%	-4,48%	10,17%	25,43%	
112648	1	1,56%	-4,15%	10,45%	25,67%	
111060	1	1,56%	-2,68%	11,72%	26,72%	

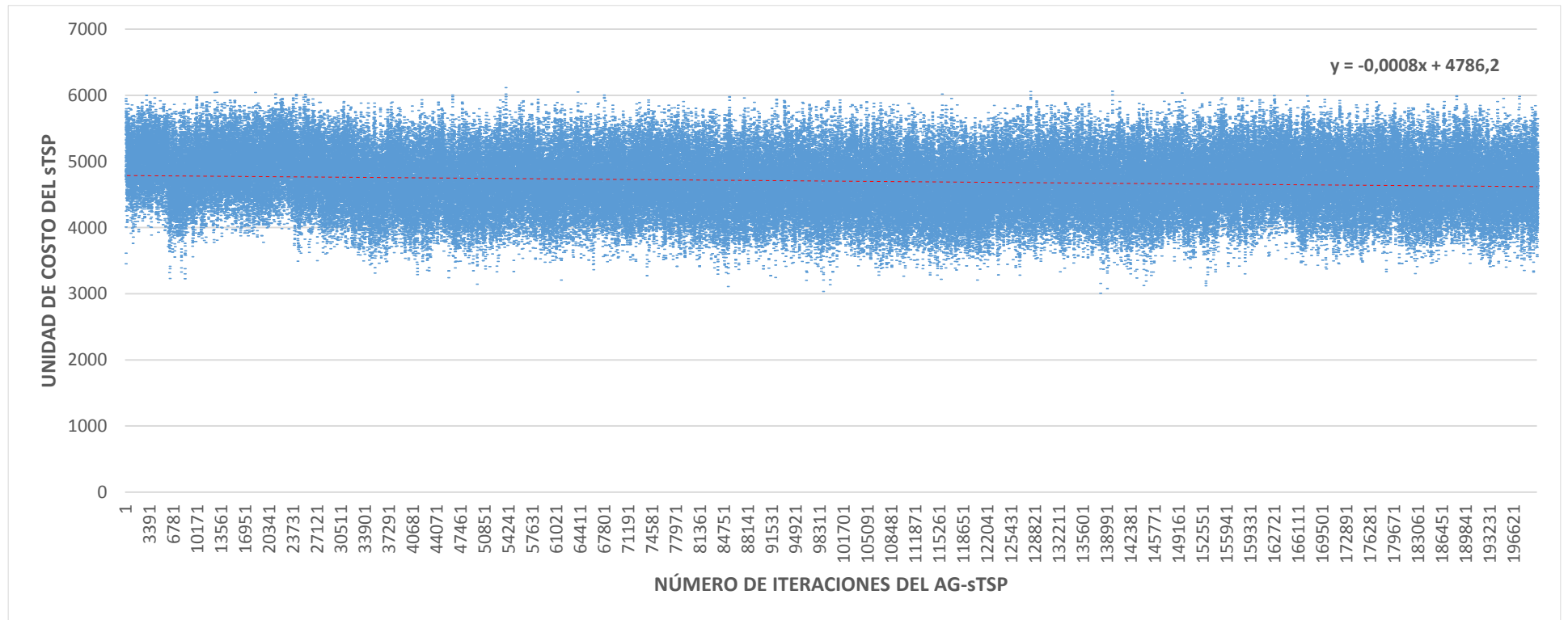
Fuente: Elaborado por el Autor.

## Anexo G

A continuación, en la gráfica 7 se presenta a manera de ejemplo, los resultados de la corrida 1 de la réplica 1 de la sesión bays29 (09), en la que se puede evidenciar la aplicabilidad del concepto de población inadaptada en la ayuda por controlar y mejorar la variabilidad de la adaptabilidad promedio de las poblaciones concurrentes de soluciones.

Remítase a la carpeta de Anexo "00 Tabulación de Resultados AG para Instancias sTSP-TSPLIB", desde el archivo 06 de extensión ".xlsx", para una consulta completa del procesamiento de la corrida 1 de la réplica 1 de la sesión bays29 (09).

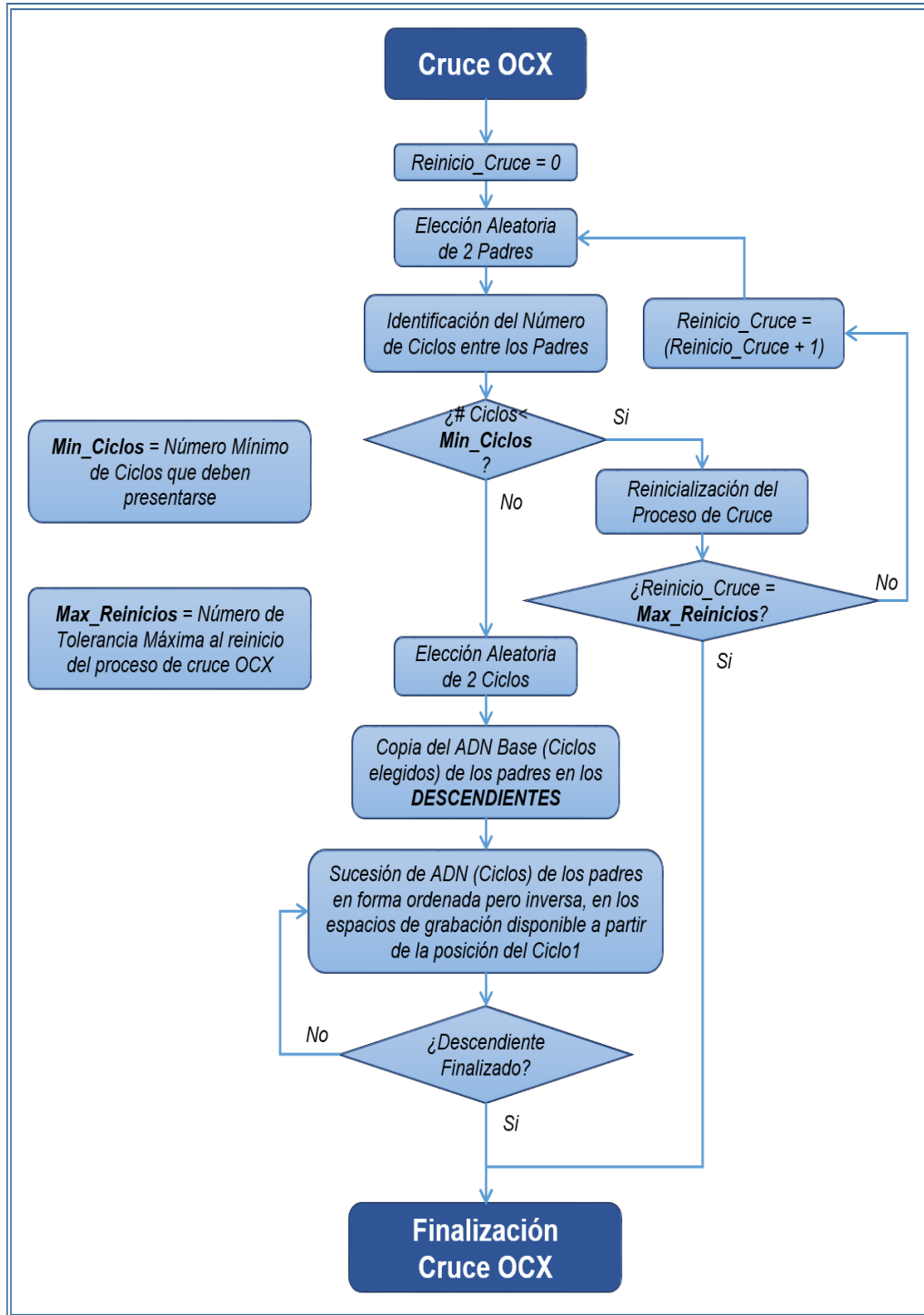
**Gráfica 7. Resultados de la Corrida 1, Réplica 1 de la Sesión bays29 (09).**



Fuente: Elaborado por el Autor.

Anexo H

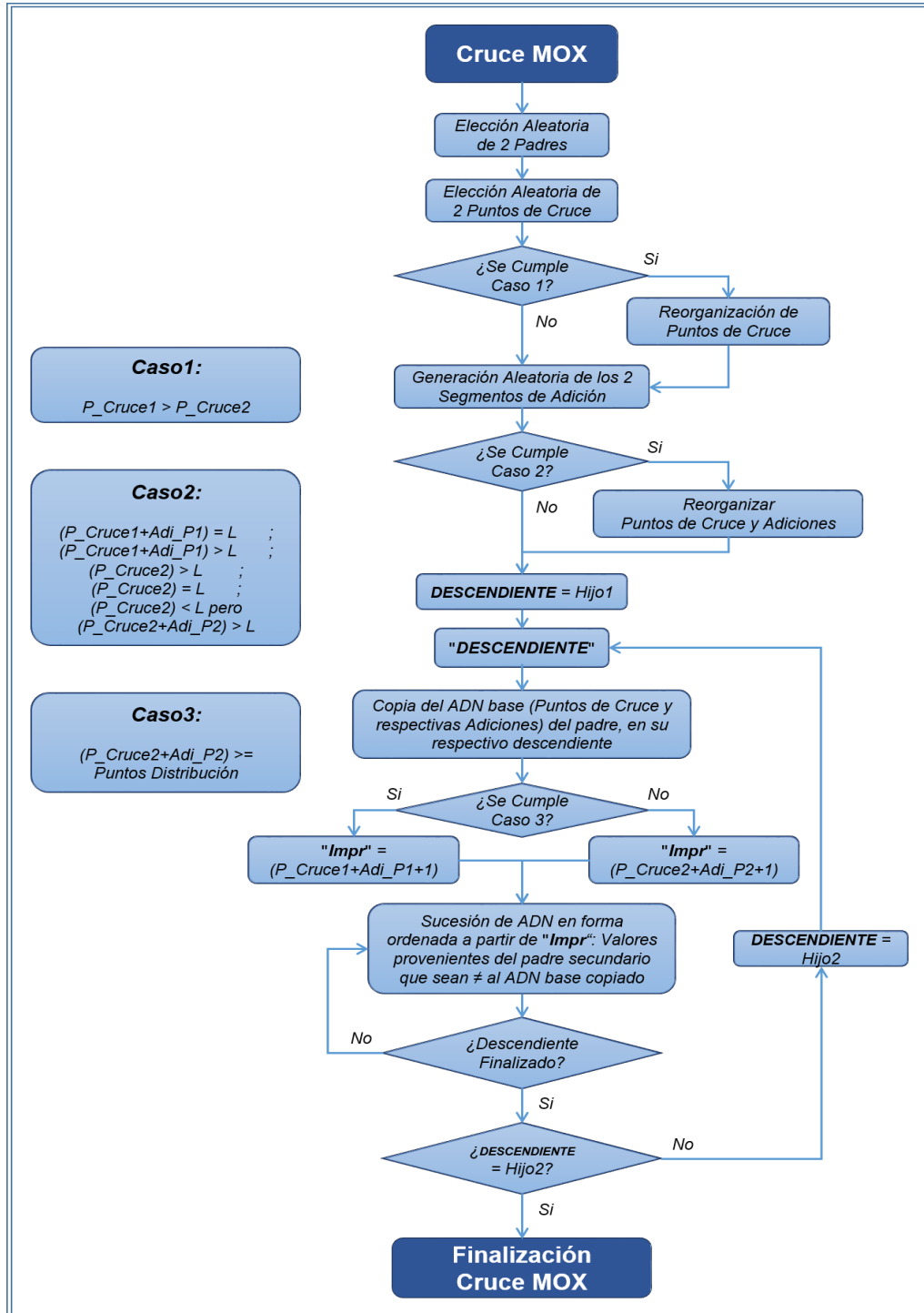
Ilustración 18. Pseudocódigo Gral. del operador de cruce propuesto "OCX".



Fuente: Elaborado por el Autor

Anexo I

Ilustración 19. Pseudocódigo Gral. del operador de cruce propuesto "MOX".



Fuente: Elaborado por el Autor